# REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188



| 1. AGENCY USE ONLY (Leave blank) | 2. REPORT DATE | 3. REPORT TYPE AND DATES COVERED |
|---|---|---|
| | DECEMBER 1990 | Special Report Jul 88 to Jun 89 |

**4. TITLE AND SUBTITLE**

Design of a Decoupling Controller for the UTAH/M.I.T. Dexterous Robot Hand

**5. FUNDING NUMBERS**

723138D3

**6. AUTHOR(S)**

Sridhar Adapalli

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**

Harry G. Armstrong Aerospace Medical Research Laboratory
Wright-Patterson AFB OH 45433-6573

**8. PERFORMING ORGANIZATION REPORT NUMBER**

AAMRL-SR-90-518

**9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)**

N/A

**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**

N/A

**11. SUPPLEMENTARY NOTES**

N/A

**12a. DISTRIBUTION/AVAILABILITY STATEMENT**

Approved for Public Release; distribution is Unlimited

**12b. DISTRIBUTION CODE**

**13. ABSTRACT** *(Maximum 200 words)*

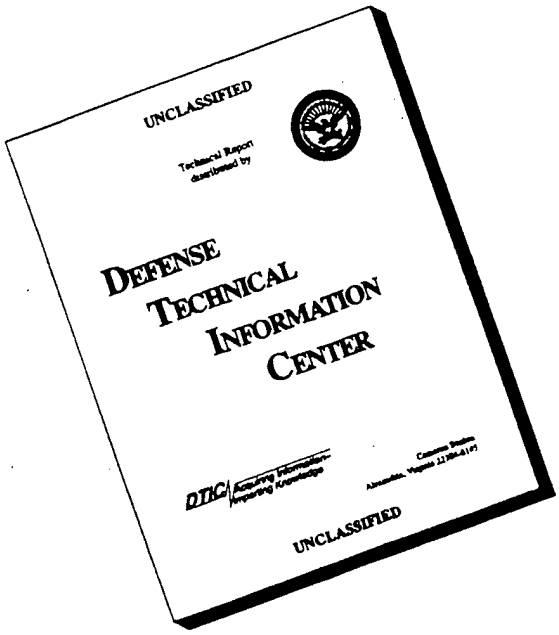
This thesis describes the design of a decoupling controller for the Utah/MIT Dexterous Hand (DH). The DH is of a semi-anthropomorphic configuration and possesses three fingers and a thumb, each consisting of four joints. An analog controller of the proportional plus derivative type, is used to control the movements of different joints of the DH.

The different gain blocks of the analog controller and the position and tendon tension sensors of each joint are calibrated. The dynamics of each joint is determined, and the control parameters for the PD controller of each joint are calculated. The three distal joints of the DH are mechanically coupled, i.e. the movement of one joint results in the movement of the other coupled joints. An algorithm to decouple different joints of the DH is proposed. The simulation results obtained using SYSTEM BUILD utility of MATRIX$_x$ show that joints of the DH can be controlled independently when the decoupling algorithm is included with the PD controller. Software, to implement the decoupling algorithm, is written in the 'C' programming language.

| 14. SUBJECT TERMS | 15. NUMBER OF PAGES |
|---|---|
| | 16. PRICE CODE |

| 17. SECURITY CLASSIFICATION OF REPORT | 18. SECURITY CLASSIFICATION OF THIS PAGE | 19. SECURITY CLASSIFICATION OF ABSTRACT | 20. LIMITATION OF ABSTRACT |
|---|---|---|---|
| Unclassified | Unclassified | Unclassified | Unlimited |

# DISCLAIMER NOTICE




**THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.**


AAMRL-SR-90-518

# DESIGN OF A DECOUPLING CONTROLLER FOR THE UTAH/M.I.T. DEXTEROUS ROBOT HAND

Sridhar Adapalli

WRIGHT STATE UNIVERSITY
DEPARTMENT OF ELECTRICAL ENGINEERING
DAYTON, OHIO 45431

DECEMBER 1990

FINAL REPORT FOR THE PERIOD JULY 1988 TO JUNE 1989



HARRY G. ARMSTRONG AEROSPACE MEDICAL RESEARCH LABORATORY
HUMAN SYSTEMS DIVISION
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OH 45433-6573

## NOTICES

When US Government drawings, specifications or other data are used for any purpose other than a definitely related Government procurement operation, the Government thereby incurs no responsibility nor any obligation whatsoever, and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise, as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

Federal Government agencies registered with Defense Technical Information Center should direct requests for copies of this report to:

Defense Technical Information Center
Cameron Station
Alexandria, Virginia 22314

## TECHNICAL REVIEW AND APPROVAL

AAMRL-TR-90-518

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

JAMES W. BRINKLEY
Director
Biodynamics and Bioengineering Division
Armstrong Aerospace Medical Research Laboratory

# DESIGN OF A DECOUPLING CONTROLLER FOR THE UTAH/M.I.T. DEXTEROUS ROBOT HAND

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science

By

SRIDHAR ADAPALLI
B. E., Annamalai University, India, 1986

1989
Wright State University

# ABSTRACT

Adapalli, Sridhar M.S., Department of Electrical Engineering, Wright State University, 1989. Design of a Decoupling Controller for the Utah/M.I.T. Dexterous Robot Hand.

This thesis describes the design of a decoupling controller for the Utah/M.I.T. Dexterous Hand (DH). The DH is of a semi-anthropomorphic configuration and possesses three fingers and a thumb, each consisting of four joints. An analog controller of the proportional plus derivative type, is used to control the movements of different joints of the DH.

The different gain blocks of the analog controller and the position and tendon tension sensors of each joint are calibrated. The dynamics of each joint is determined, and the control parameters for the PD controller of each joint are calculated. The three distal joints of the DH are mechanically coupled, i.e. the movement of one joint results in the movement of the other coupled joints. An algorithm to decouple different joints of the DH is proposed . The simulation results obtained using SYSTEM_BUILD utility of MATRIX$_x$ show that joints of the DH can be controlled independently when the decoupling algorithm is included with the PD controller. Software, to implement the decoupling algorithm, is written in the 'C' programming language.

# TABLE OF CONTENTS

APPENDICES

# LIST OF FIGURES

# LIST OF TABLES

# ACKNOWLEDGEMENT

This project was supported by the Armstrong Aerospace Medical Research Laboratory at Wright-Patterson Air Force Base, Dayton, Ohio.

I would like to acknowledge and thank my thesis advisor Dr. Kuldip S. Rattan for his time, guidance and patience, without whose help the completion of this thesis would have been a distant dream, Dr. Ints Kaleps of Wright-Patterson Air Force Base - AAMRL and Dr. Russel Hannen for their critical comments, Dr. Daniel W. Repperger and Dr. Pradeep Misra for their valuable advice.

My special thanks to Mr. Ken Johnston, Mr. Monty Crabill and Mr. Jeff Logan of Systems Research Laboratory for their invaluable support in helping me complete this project successfully.

I would also like to acknowledge the help provided by my officemates Ranvir Solanki and Khalid Barazanji, the staff members of Electrical Engineering Department Offices especially Ann Isper, Suzy McGovern and Serena Tan for their help whenever I needed.

This thesis is dedicated to my friends for their understanding, patience and encouragement while this work was being done.

# I . INTRODUCTION

## 1.1 AN OVERVIEW

The Air Force has a great need to maintain force survivability and base operability during wartime scenarios in chemical, biological and radiological environments. The Robotic Telepresence (R/T) program at the Harry G. Armstrong Aerospace Medical Research Laboratory (AAMRL) is based on the need to project human intelligence, perceptual capabilities and motor skills into hostile environments through the use of human controlled robots, thereby removing the human operator from hazardous environments. One of the objectives of the R/T program is to investigate the feasibility of utilizing remote human-in-the-loop control of dexterous robots to perform operational tasks such as aircraft inspection and servicing, explosive ordinance disposal, and environmental monitoring and decontamination [1].

In the R/T concept, human-in-the-loop control of the remote robot is achieved by feeding robot/task state information back to the operator via intuitive and natural displays. This allows the operator to effectively 'see, feel and manipulate' objects at the work-site and then issue appropriate position/force commands to the slave. The control system will link human judgement, adaptability and dexterity in real time to a robot system capable of operating in lethal environments. In some cases, the human-controlled robot performance can be scaled up or down to substantially ex-

ceed human capabilities, such as lifting heavy objects or performing micro movements with precision end effectors.

The goal of the R/T program is to develop a series of dynamic telepresence test cells integrating state-of-the-art components such as visual displays, sensors, and robotic manipulators. These test cell stations will provide experimental tools for research into human factor issues associated with the human-in-the-loop control of tele-robotic systems. Issues to be investigated include master/slave control of dexterous robotic arms and hands, force reflection techniques, time delay compensation, visual and tactile feedback requirements, etc. Development and implementation of appropriate control algorithms are required to allow human-in-the-loop control of the remote system. The remote system pertaining to this report is the Utah/M.I.T. dexterous robot left and right hands attached to appropriate robot arms.

## 1.2 SCOPE OF THE R/T PROGRAM

One of the objectives of the R/T program is to develop and implement dexterous hand control algorithms to effectively slave the Utah/M.I.T. Dextrous Hand (DH) to an exoskeleton master worn by the human operator. A right and left pair of Utah/M.I.T. Dexterous Hands has been installed at AAMRL at Wright Patterson Air Force Base.

Figure 1.1 depicts the master/slave dexterous hand system architecture [1]. The position sensors in the exoskeleton master send position information to the master processor. The master processor converts these positions to normalized human joint

positions. The host computer maps the normalized human joint positions to the desired joint positions of the slave, and issues appropriate position/force commands to the slave. The joints on each digit of the dexterous hand are mechanically coupled, i.e. when one joint is commanded to move, the other joints also move. To overcome this, the host computer, with the help of decoupling algorithms, decouples the different joints of the hand. The host computer then issues appropriate position/force commands to the analog servo controller of the DH. The analog controller applies a current that is proportional to the input voltage to the pneumatic actuators. These in turn, drive the joints which are connected to the actuators through tendons. The joint position and tendon tension data are fed back to the analog controller. A part of the above mentioned work is being performed at Wright State University (shown enclosed in dotted lines in figure 1.1). This report deals with the development of the the decoupling hand control algorithms and the design of the individual loop joint design parameters required to slave the Utah/M.I.T. DH to the exoskeleton master.

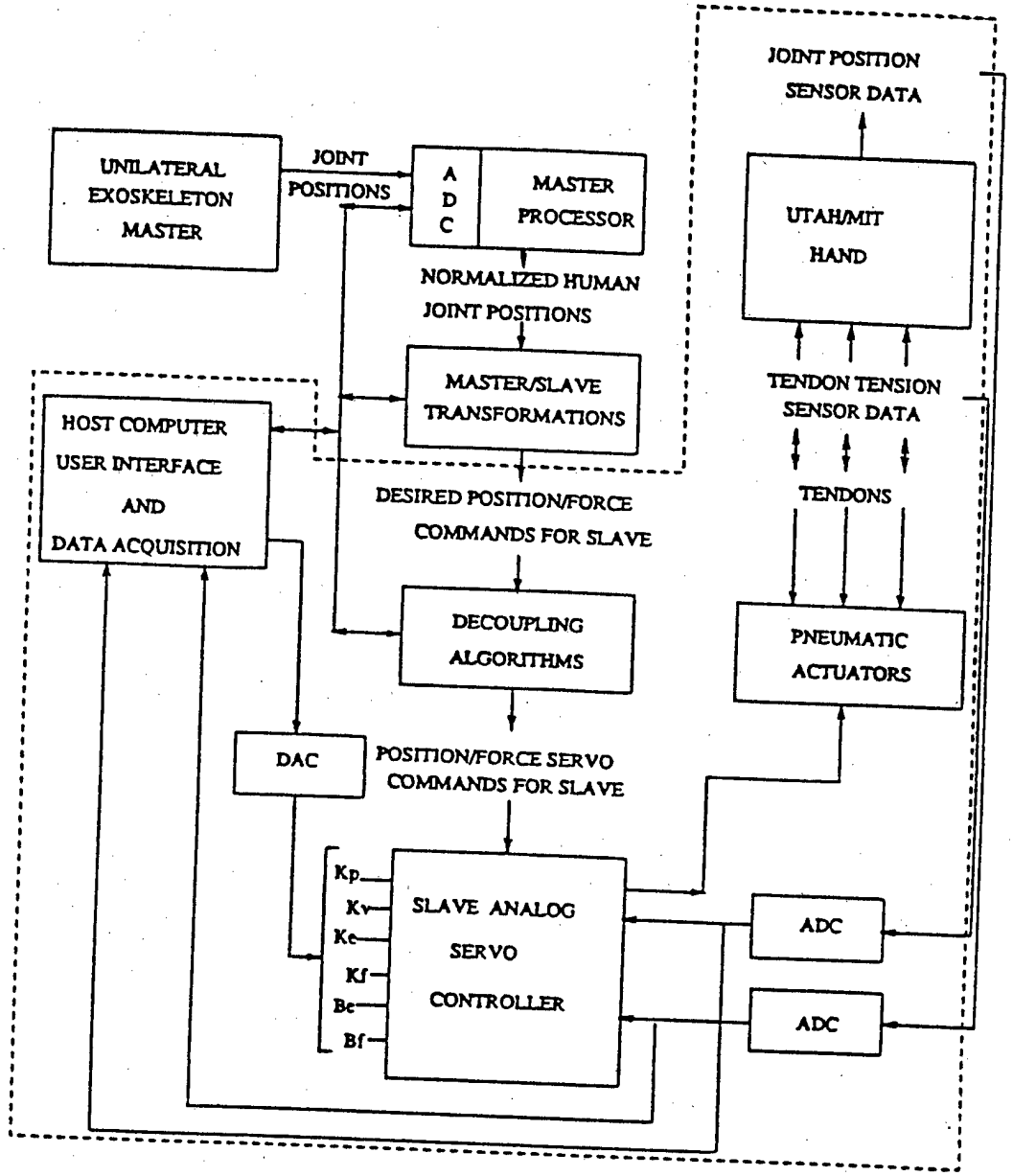



Figure 1.1 Master/slave dexterous hand system architecture [1]

## 1.3 THESIS OBJECTIVE

The objective of this project is to

- experimentally measure and perform computer simulations of the operating characteristics of the different joints of the Utah/M.I.T. DH.
- identify the transfer function of each joint of finger 1 of the Utah/M.I.T. DH.
- experimentally determine the design parameters for the position control loop of each individual joint.
- develop a scheme to decouple different joints of a finger.
- perform computer simulations of the decoupling algorithms in $MATRIX_x$
- develop a C code for the decoupling algorithms.

# II. BACKGROUND

This chapter discusses the need for dexterous end effectors and describes the Utah/M.I.T. dexterous hand.

## 2.1 NEED FOR DEXTEROUS END EFFECTORS

Research in dexterous hands for manipulators is a relatively new area of robotics, which is now developing rapidly. This is because of the increasing realization among researchers that dexterous end effectors have a wide range of applications in industry. The lack of capabilities of current end effectors is increasingly being felt as one of the main reasons for the limited usage of robots in current industrial applications. The study of dexterous end effectors is an exciting field with many unsolved problems in both mechanical design and control [2].

Dexterous end effectors with sensing abilities need to be developed for the following reasons :

1. Current end effectors limit the flexibility and capability of manipulator systems due to lack of ability to handle arbitrary parts. The most common end effector currently being used in industry is the parallel jaw gripper. The parallel jaw gripper, while capable of performing a number of tasks, nevertheless limits the kind of objects which may be grasped and the ways in which they may be

manipulated. The use of parallel jaw grippers in industry can be compared to a car mechanic using only pliers to execute all repairs [3].

2. Current end effectors also lack sensing abilities. The vast majority of current robotic applications in industry involve applications such as spot welding and spray painting tasks, which do not involve contact of the end effector with the environment. These tasks do not require precise positioning of the robots and can be easily handled by current industrial robots. But if robots are to be used in operations such as carrying an odd shaped part, dexterous end effectors with sensing abilities are needed.

3. In assembly operations, manipulator end effectors are required to perform small incremental operations for precise parts mating. Due to mechanical and control considerations, small incremental movements are not possible with large segments of present day industrial robots. But with the help of dexterous end effectors, robots can be used to perform assembly operations which require a lot of precision without the involvement of large segments of the manipulator. The large segments of the manipulator can be used to bring the end effector into the vicinity of the assembly operation, which would then be completed by precision movements with the dexterous end effector.

4. Just as special purpose hands are currently needed to permit different operations by the manipulator, a special purpose fixture is required for holding a particular part. With a dexterous hand capability, a manipulator could serve as an adaptable and active fixture as well. Two manipulators could then work

together in an assembly operation, one holding a part while the other performs an operation on it [2].

With a dexterous hand capability, robots could be upgraded from stereotyped situations such as assembly lines. Non-assembly line operations such as maintenance and repair would also become feasible.

Due to the above-mentioned considerations, researchers at the University of Utah and Massachussetts Institute of Technology felt the need to design and build a dexterous hand for research purposes. Their primary objective was to design and construct a general purpose research tool for the study of machine-based manipulation. From the outset, the emphasis was on producing a system with extraordinary performance (both active and passive) and factors related to cost, power consumption and other practical considerations were de-emphasized. The device was not intended for near-term application in an industrial environment. Rather it was intended to be a research tool with sufficient functional capabilities to permit a broad variety of experiments. This would enable researchers to understand fundamental concepts and machine design issues involved in the study of dexterous end effectors. After several years of effort, the Center for Biomedical Design at the University of Utah and The Artificial Intelligence Laboratory at the Massachussetts Institute Of Technology (M.I.T.) designed and developed an anthropomorphic dexterous robot hand, which they called 'The **UTAH/M.I.T. DEXTEROUS HAND**. The Dexterous Hand (DH) was designed as a series of modules which could be interchanged for maintenance purposes, permitting reconfiguration of the system into alternate geometries [4].

## 2.2 DESCRIPTION OF THE UTAH/M.I.T. HAND

The Utah/M.I.T. hand is of semi-anthropomorphic size and design. The photograph of the Utah/M.I.T. Dexterous Hand and the line diagram of the same are shown in figures 2.1 - 2.5 .

The choice of semi-anthropomorphic (almost human-like) geometry for the hand is due to the following three reasons :

1. The human hand is existing proof that a wide variety of very complicated, manipulative tasks can be accomplished by a single system, provided that the end effector includes sufficient complexity to address proposed tasks and that the control system and its sensors are adequate.

2. A semi-anthropomorphic configuration allows the researcher to compare the operations of the robot hand with operations of his own natural hand.

3. A semi-anthropomorphic configuration has potential application as a slave element in a tele-operation system.

The hand consists of

1. Three four degree-of-freedom fingers.

2. One four degree-of-freedom thumb.

3. One three degree-of-freedom wrist.

The three fingers and the thumb consist of three parallel axes joints, which provide the curling action. The fourth joint is the proximal joint which is perpendicular to

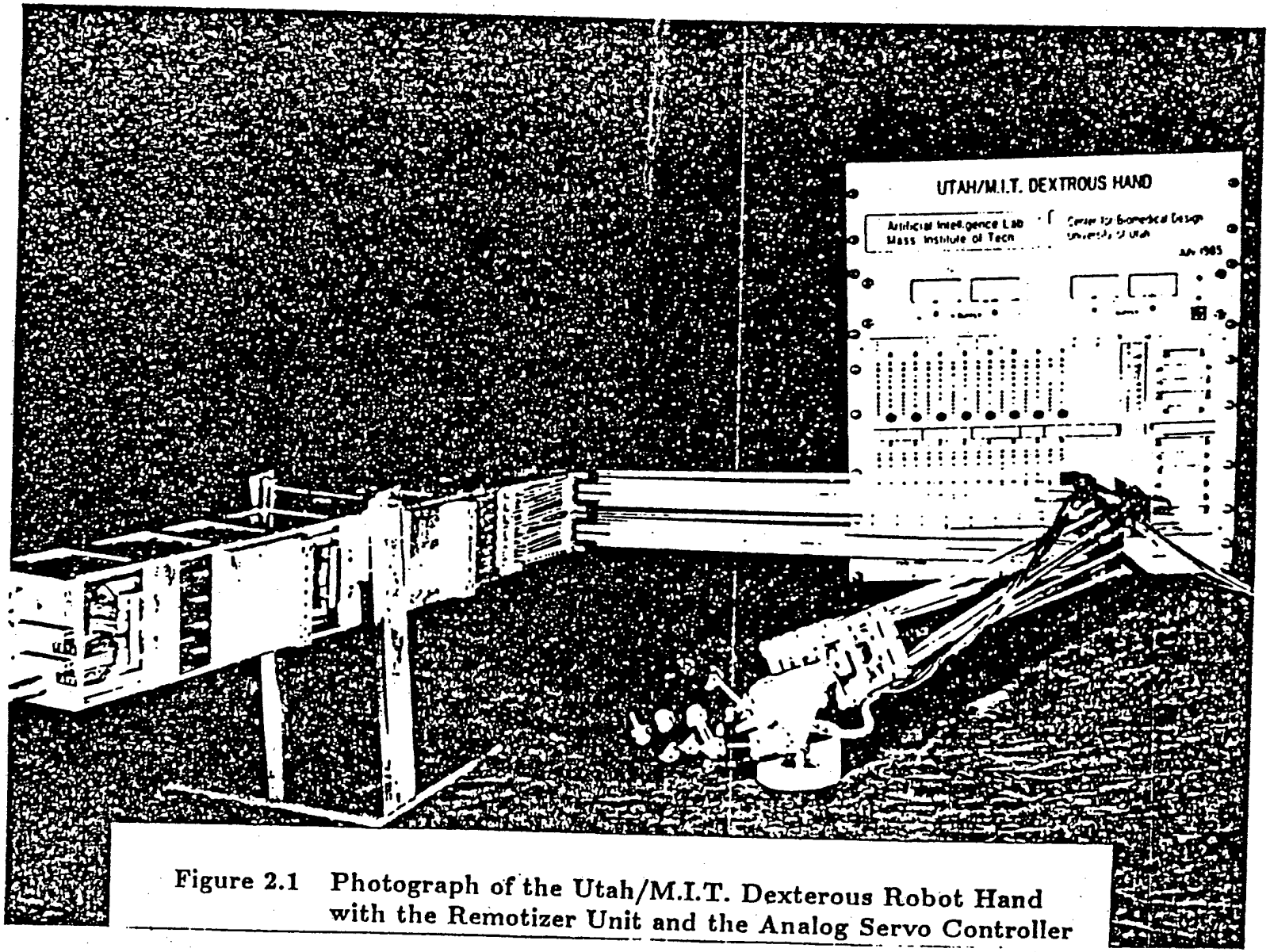



**Figure 2.1** Photograph of the Utah/M.I.T. Dexterous Robot Hand with the Remotizer Unit and the Analog Servo Controller

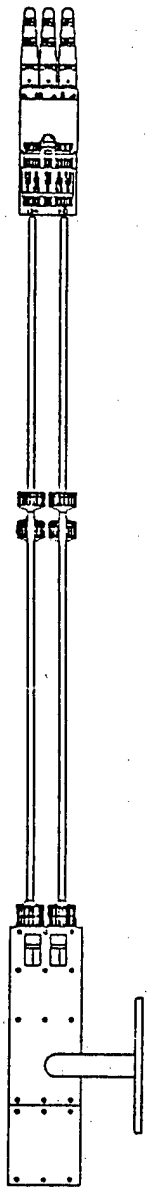


Figure 2.2 Top view of the Utah/M.I.T. DH with remotizer and actuation system

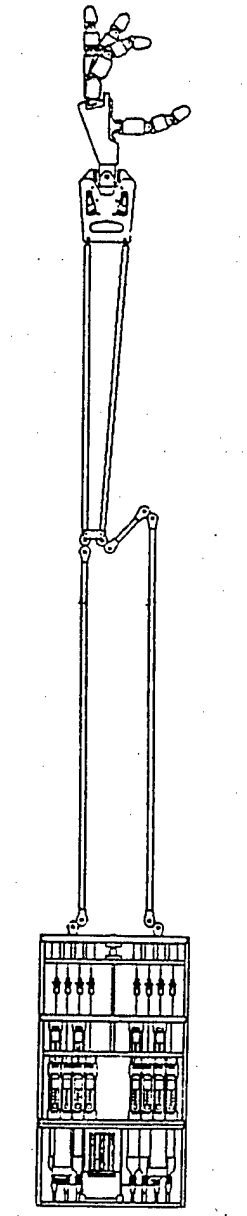


Figure 2.3 Side view of the Utah/M.I.T. DH with remotizer and actuation system

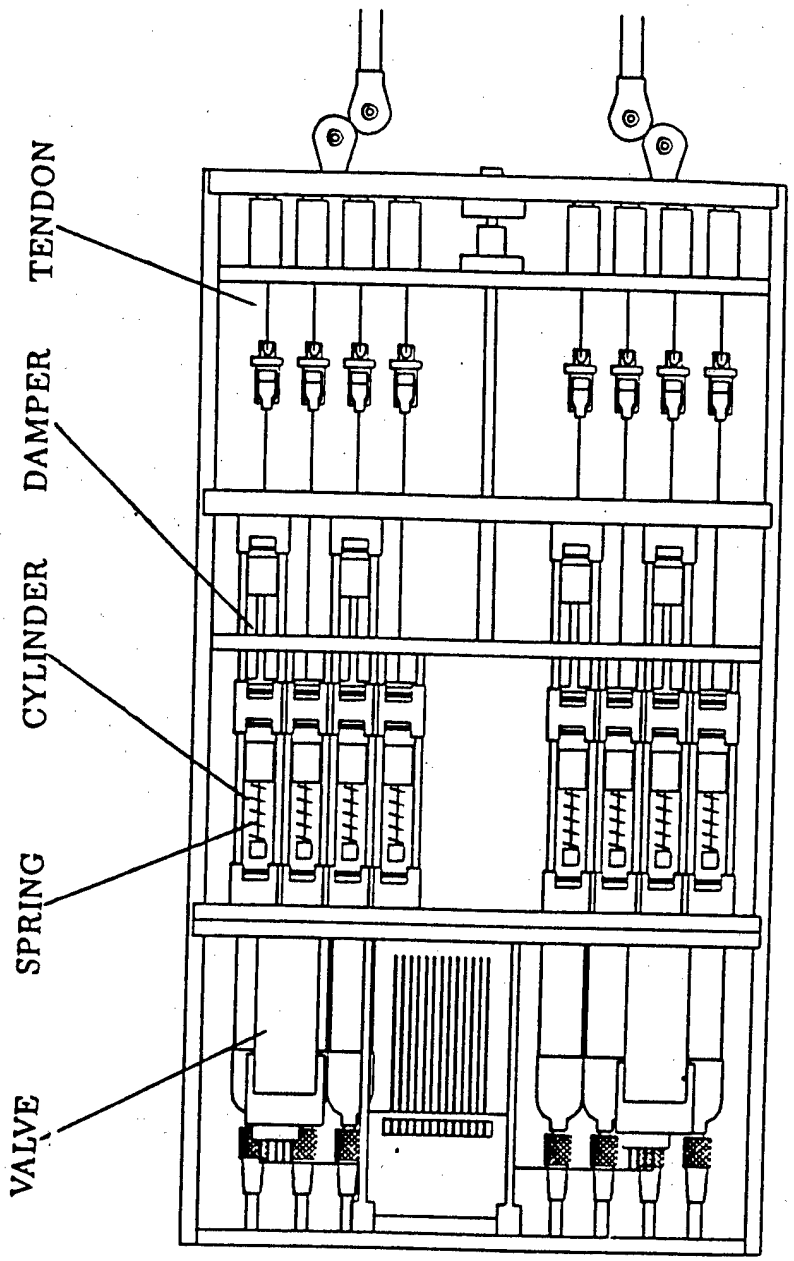



**Figure 2.4** Line drawing of the actuation system of the Utah/M.I.T. DH

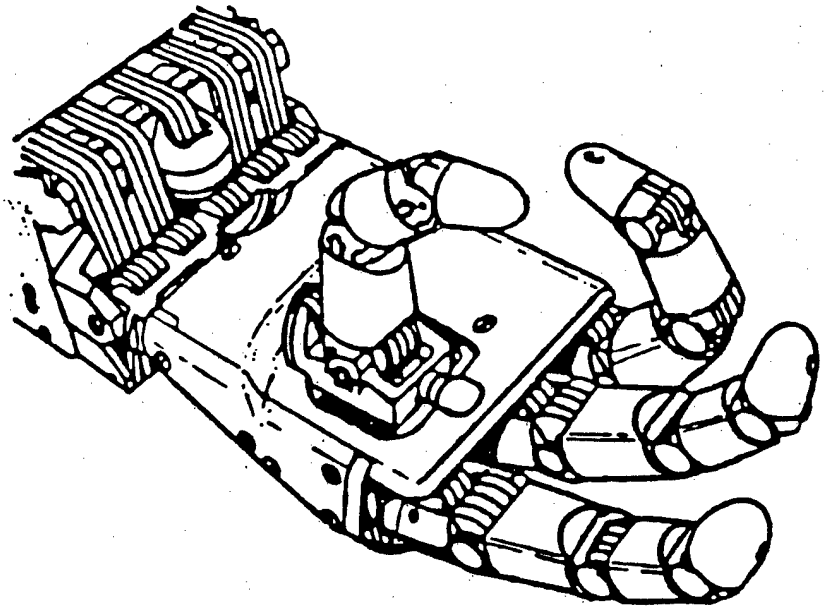


Figure 2.5 Line drawing of the Utah/M.I.T. Dexterous Robot Hand [4]

the other joints (figure 2.6). This joint provides side-to-side motion (roll).

## 2.3 COMPARISON WITH THE HUMAN HAND

The Utah/M.I.T. hand differs from the human hand in the following ways :

1. The Utah/M.I.T. hand has three fingers and a thumb, while the human hand has four fingers and one thumb. This is to reduce the complexity of tendon routing and computational complexities.

2. The first two joints of each digit (the 0 and 1 joints), shown in figure 2.6, are separated in order to allow tendons to be routed in a manner which results in a reliable operation.

3. The axis of joint 0 is at an angle of 12° to the base plane (figure 2.6). In the human hand, the $0^{th}$ joint axis is essentially perpendicular to the base plane. This difference is primarily due to tendon routing limitations. Also, this allows the fingers to achieve significant side-to-side movements (0 joint mobility) when the joint 1 is flexed to the 90 degrees position. This is in contrast to the situation found in the human hand where 0 joint mobility is almost null when joint 1 is flexed to 90 degrees.

4. The fingers and thumb of the Utah/M.I.T. hand have a side-to-side motion, in contrast to the abduction/adduction motion of the human hand.

5. The thumb of the Utah/M.I.T. hand is placed on the palmar section of the hand between the first and second fingers, which is unlike the human hand.

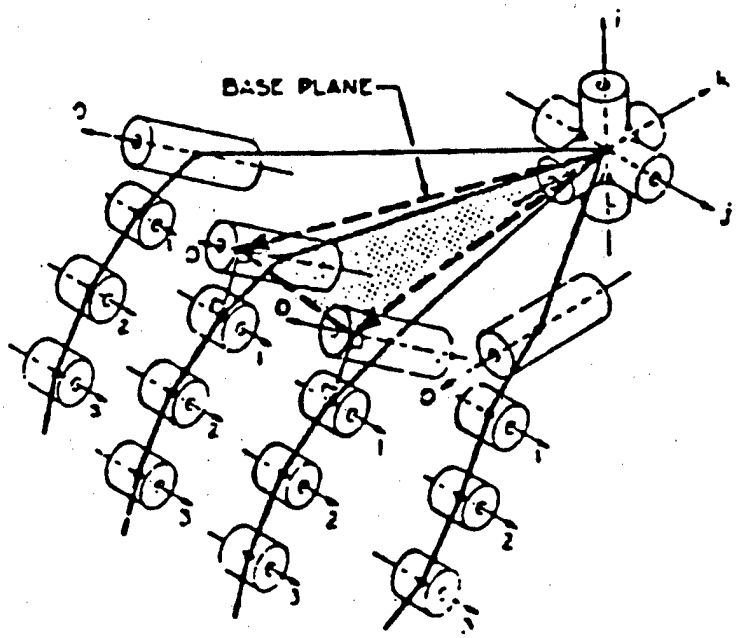



Figure 2.6 Configuration of the Dexterous Hand showing the orientation of each joint [4]

This allows tendons to be routed over the wrist and through the palm to the thumb in a reliable manner.

6. The wrist joint is larger than the human hand. This is again due to tendon routing problems.

The hand is connected to the wrist, which is attached to the remotizer unit. The remotizer carries the tendons from the actuator package, which is maintained in a static position by an external mounting.

## 2.4 TENDONS

Each joint is driven by two tendons, thus requiring '2n' tendons to drive 'n' joints. While tendon routing became complex, the decoupling between the joints is made simple. The tendons are composite structures which consist of Dacron fibers woven around multiple longitudinal elements of Kevlar. The Dacron outer sheath serves to align and protect the internal load bearing Kevlar fibers. Tendons provide a number of advantages including very low mass, and the possibility for stiff transmission of energy over complex pathways. Tendons require no bulky terminal energy transformation systems such as cylinders used in hydraulic systems and motors with transmissions used in electrical systems. Tendons are routed throughout the hand via sequences of pulley-induced bends as shown in figure 2.8. The pulleys used have varying diameters of .95, .72, .49 and .38 inches depending on the location, with concave contact surfaces shaped in order to accomodate the cyllindrically shaped tendons. One side of the tendon is attached to the piston in the actuator package by adapt-

ing the overwrapped friction lock method (A) as shown in figure 2.7. The tendon is locked via a supplemental metal screw. The other side is attached to a hard stop at the joint (C) as shown in figure 2.7.

## 2.5 REMOTIZER

The DH was initially designed to be positioned in space by a manipulator arm. However, due to the limited capabilities of the then existing robots, it was unlikely that the entire DH and its actuation package could be accurately and quickly moved around in space. Hence it was decided to construct a remotizing system which would enable the actuation package to remain in a static position with respect to the DH. The remotizing system includes 32 tendon pathways in four subsystems. Each of these subsystems include series of longitudinal beams and rolling joints (figure 2.9). The longitudinal beams support the compressive stresses imposed by tendons and the system of rolling joints permit motion of the remotizer without altering tendon path lengths. This configuration includes the side benefit that no torques are imposed on the remotizer joints as tendon tensions are varied. The remotizer is then a passive system which allows the DH to be freely positioned in space while the DH receives substantial energy from the actuation package [4]. The present configuration allows the DH to be positioned everywhere within a cube of sides three feet long. The total remotizer consists of four subsystems. Figure 2.9 shows the four subsystems operating in parallel to conduct 32 tendons from actuator to hand.

Each of these subsystems include the following :

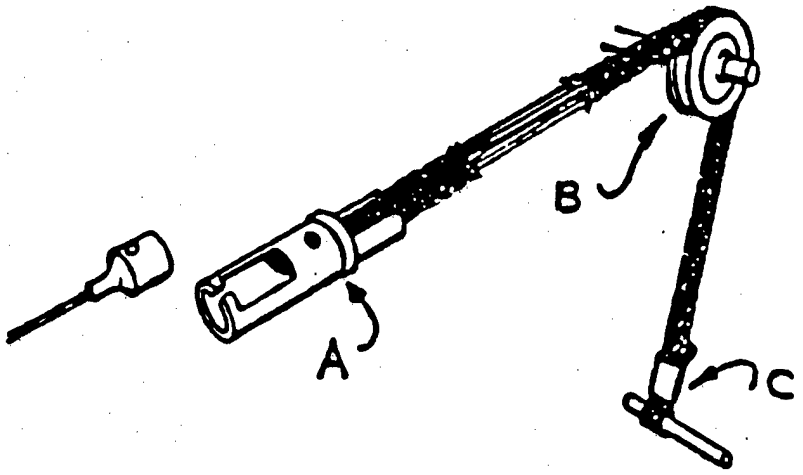



Figure 2.7 Tendons are routed over pulleys (B) and terminate either permanently as shown in point (C) or in a reversible manner shown at point (A) [4]

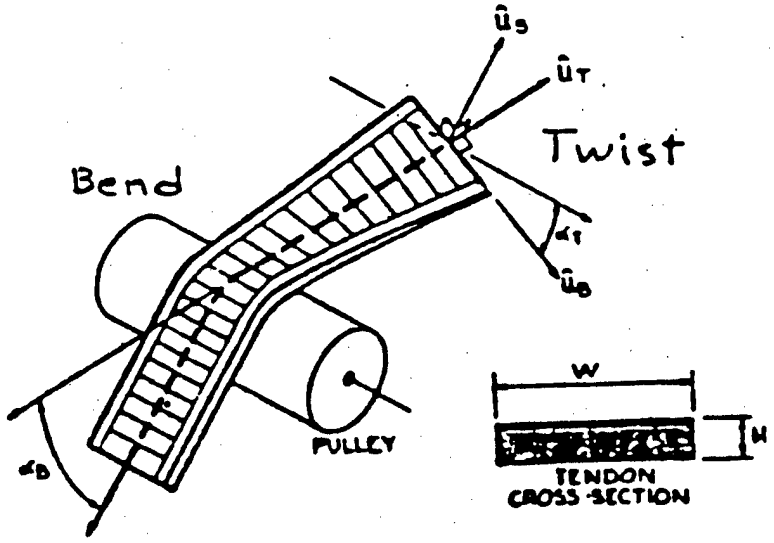



Figure 2.8 Routing of tendons throughout the system via a series of axial twists and bends over pulleys [4]

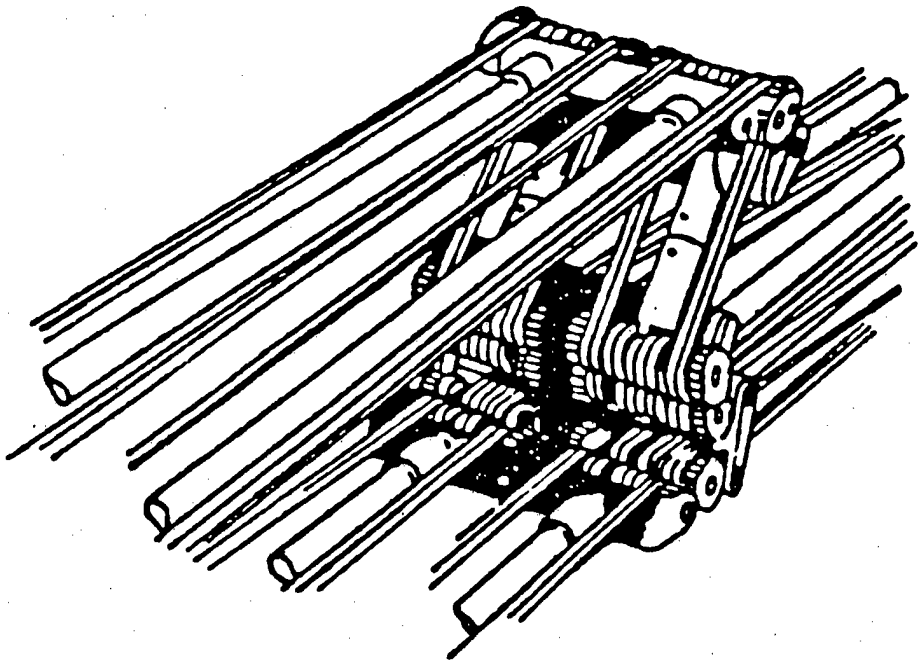


Figure 2.9 Oblique view of the central joint (elbow) of the remotizer [4]

1. Rolling joints made of injection molded composite materials into which gears have been molded. Figure 2.10 shows the side view of the central remotizer joint showing rolling action of the remotizer subsection. The orientation of the two halves of the joint is maintained by coupling members and the gears which constrain the joint to roll in a manner which maintains tendon length regardless of angular deflection of the joint. Each rolling joint includes 16 pulleys for tendon routing which means that the entire remotizer utilizes 288 pulleys from the actuation package up to tendon tension sensors in the wrist.

2. Compression rods which counteract tendon tensions and allow axial rotation in order to add degrees of freedom to the remotizer assembly.

The four subsystems must be flexibly attached in such a way as to maintain their general coordination while permitting simultaneous relative motion as the remotizing system is bent and twisted [4].

The advantages gained by using tendons and the remotizer are twofold :

1. Less weight : The hand itself can now be made very compact and light.

2. Bigger Workspace: Since the hand is attached to the remotizer unit, the hand's workspace is supplemented by that of the remotizer, thus increasing the workspace of the hand.

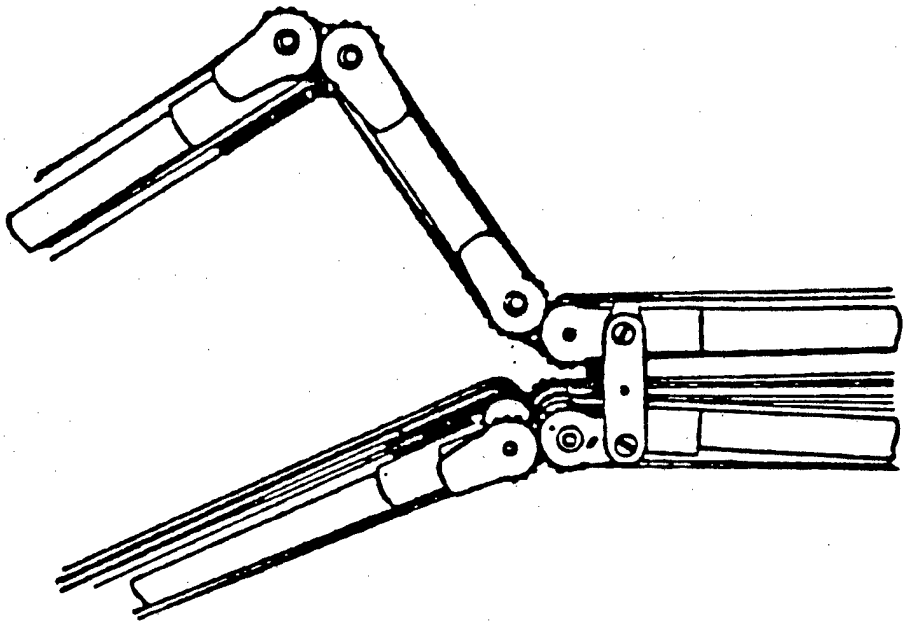


Figure 2.10 Side view of the central remotizer joint [4]

## 2.6 THE ACTUATION SYSTEM

The actuation system for the joints of the Utah/M.I.T. dexterous hand is of the pneumatic type. This allows for a low weight, compact actuator that can generate required speeds and forces. As shown in figure 2.11, each joint receives two tendons, driven by actuators which consist of pressure controlling valves and low stiction cylinders. There are 38 actuators in all, two for each joint. As shown in figure 2.12, each module contains two valves (A), two actuator cylinders (B), and two adjustable pneumatic dampers (C). Each valve receives an electric signal which generates a cylinder pressure. Spring tensioning systems are included within cylinders to maintain tendon tensions when the system is unpressurized. The cylinders for actuation and damping are constructed of glass and close fitting graphite pistons. This configuration operating at low pressures (50-100 psi) permits the elimination of seals, thereby minimizing stiction effects. The dampers which operate in a pressurized mode, in order to increase damping and minimize compressibility effects, are adjustable in order to allow a variable trade-off to be made between high frequency operation and stability during free-space operation. The actuator modules are compactly placed in a 4.25 × 4.25 × 24 inch rectangular assembly containing air manifolds and weighing 20 pounds.

Each actuator consists of a two-stage jet pipe valve (figure 2.13). The first stage is electrically driven to provide a pressure signal to a second stage. This increase in the primary stage pressure drives the second stage upper diaphragm downward to deflect the jet pipe and increase cavity pressure. Piston cavity pressure is fed back to a lower diaphragm such that the valve behaves approximately as a pressure source

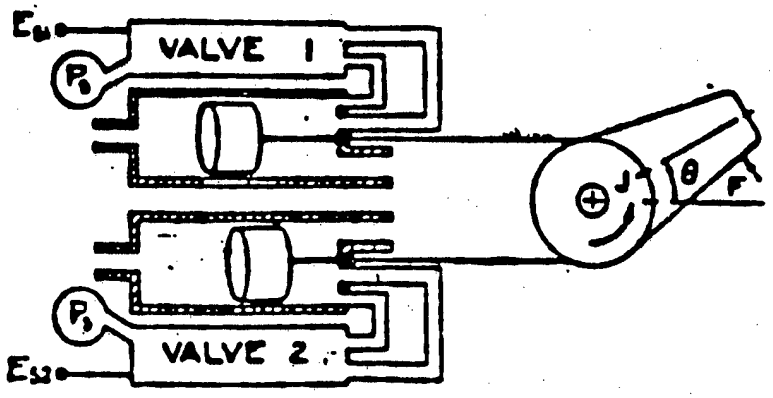



Figure 2.11 Configuration of joint and two actuators consisting of low stiction cylinders and pressure controlling valves [4]

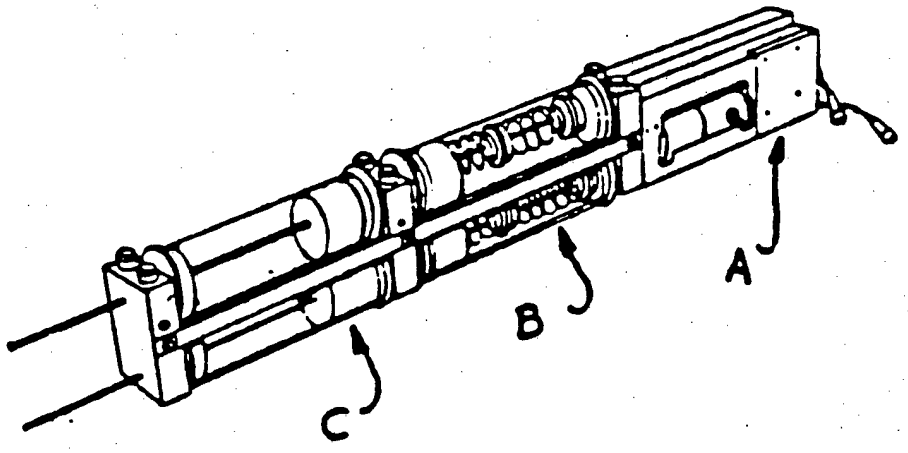



Figure 2.12 A dual actuator module which includes valves, cylinders and dampers [4]

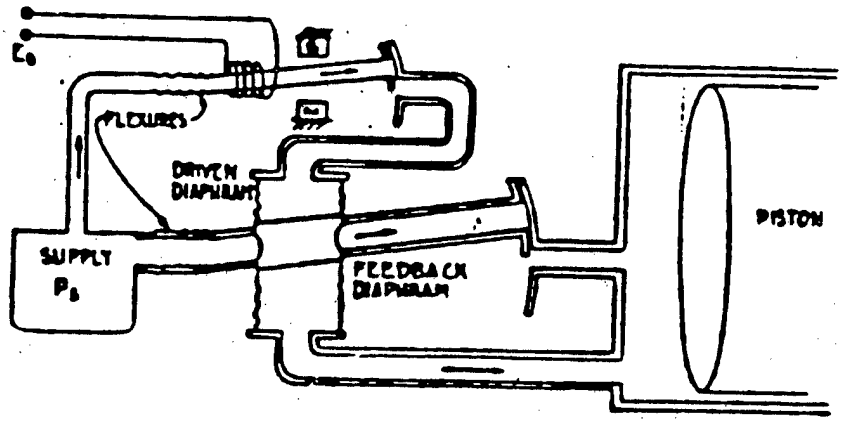



Figure 2.13 Schematic diagram showing the function of the two-stage electropneumatic, pressure contolling jet pipe valve [4]

modulated by an input current [4].

## 2.7 SENSORS

There are two kinds of sensors incorporated in the hand, namely

1. Joint Angle Sensors : The joint angle sensors are Hall-effect devices, located at the sides of each joint. These sensors provide accurate joint angle information for control purposes. These sensors are reliable, compact and proportional. The sensors produce a voltage that is proportional to the measured angular displacement. Referring to figure 2.14, (A) is a magnetically sensitive Hall effect device located in the proximal link, while (B), attached to the distal link, includes two cobalt samarium magnets operating in a dipole configuration. As the Hall effect device sweeps the magnetic field, it produces an output current which corresponds to the angular deflections between 0 and 95 degrees with a linearity within 5%. The Hall effect sensor system includes a number of advantages such as continuous operation, low friction, high bandwidth operation, no mechanical contact, long life, and tolerance to surrounding contaminants.

2. Tendon Tension Sensors : Tendon tension is monitored in order to obtain torques imposed on individual joints. This information is provided to the controller for actuator compensation. Ideally, tendon tension sensors would be installed distally at the insertion of tendons at each joint and proximally at the output of each actuator. However, due to packaging problems and other issues, a compromise system was selected. The sensors were mounted at the wrist. Each

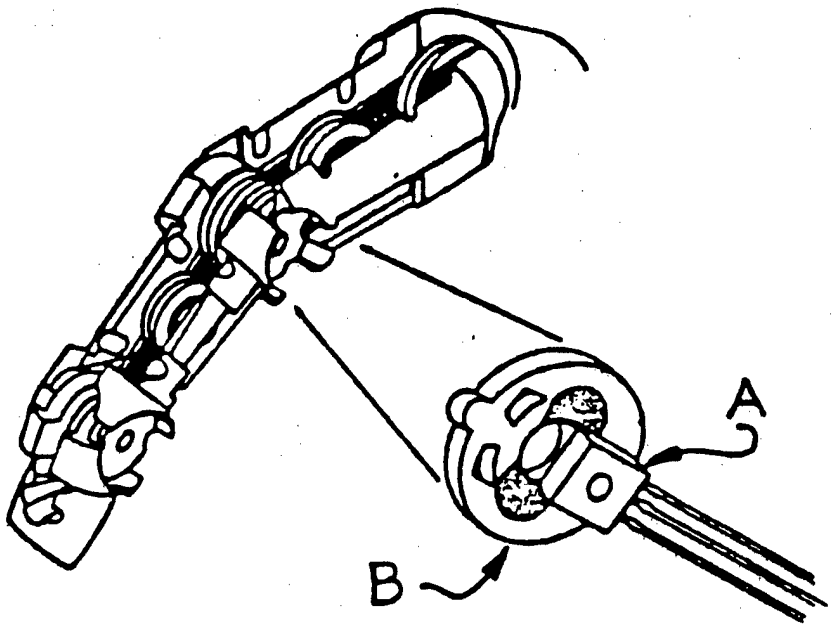



Figure 2.14 Configuration of the Hall effect sensors
used to measure angular deflection of the joints [4]

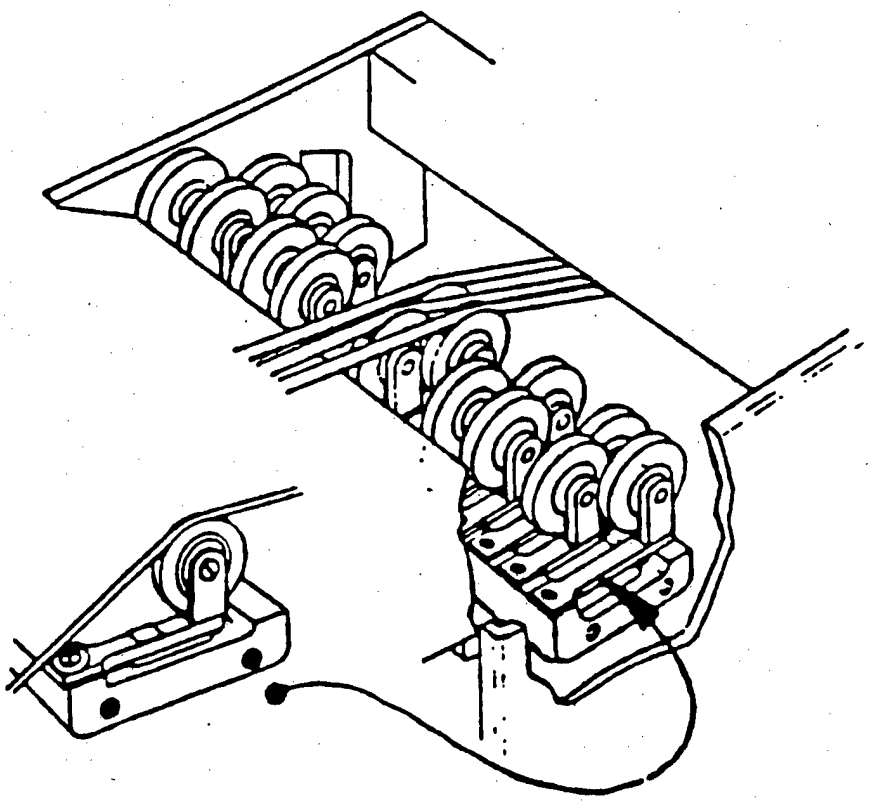


Figure 2.15 Configuration of the 32 tendon tension sensors located in the wrist of the Dexterous Hand [4]

sensor uses a semiconductor strain gauge bridge to monitor the beam deflection, which is proportional to the tendon tension. As shown in figure 2.15, the pulley is positioned in order to perturb the path of the tendon such that tendon tension imposes a load on the cantilevered beam. A semiconductor strain gauge bridge detects beam strain and provides a linear output for tendon tensions from 0 to 30 pounds [4].

The above discussion described different components of the Utah/M.I.T. Dexterous Hand. Next chapter deals with the description of the dexterous hand analog servo controller.

# III. THE DEXTEROUS HAND ANALOG SERVO CONTROLLER

Figure 2.1 shows the photograph of the DH analog servo controller. The analog controller is described in detail in the following section.

## 3.1 DESCRIPTION

The Dexterous Hand control system comprises of a 500 watt power supply, a cooling fan pack, and a card cage containing an analog based position/force servo feedback controller. The analog controller section consists of 8 servo cards, an input/output card, and 2 spare slots reserved for wrist control and master conditioning. Each of the eight servo cards comprises of 2 joint controllers. Each joint controller has its own circuitry for signal conditioning (joint angles, joint velocities, and joint tendon tensions), signal processing, and power drivers for the actuators.

The Low Level Control System (LLCS) includes in all 16 variable-loop-gain position servos to operate finger joints and 32 variable-loop-gain tension servos to modulate actuator behavior in order to control tendon tensions. Amplification and signal conditioning circuitry is also included to

1. provide current sources for driving pneumatic valves,

2. drive and monitor joint angle sensors, and

DTIC COULD NOT GET MISSING

PAGE FROM CONTRIBUTOR

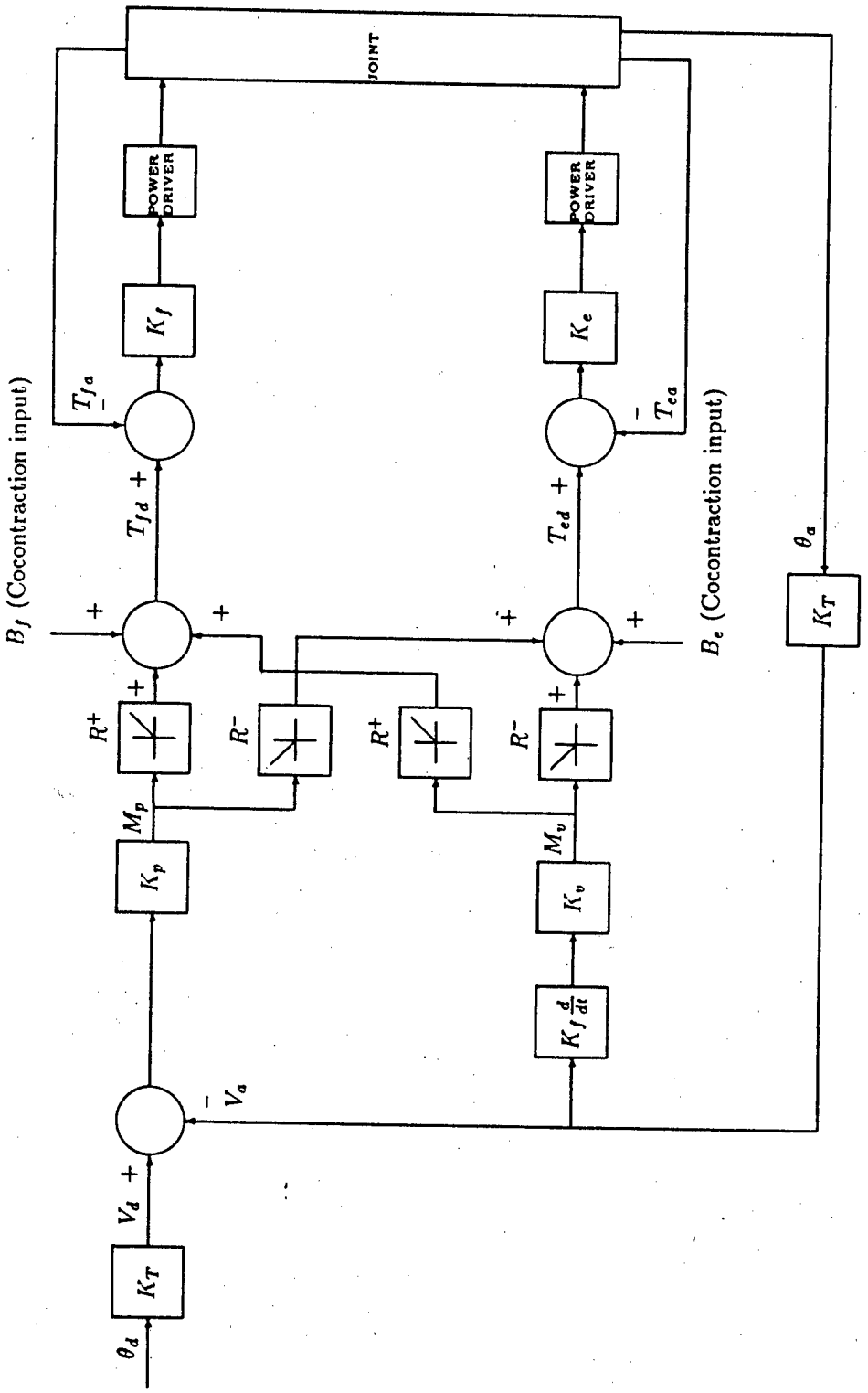



**Figure 3.2 Position control loop of a single joint**

3. drive and monitor tendon tension sensors.

The system inputs include:

1. 16 inputs for control of angular position,

2. 32 inputs for control of desired tendon tension,

3. 16 inputs to vary position servo loop gain, and

4. 32 inputs to vary tendon tension servo loop gain.

Also, a number of auxiliary inputs are available to control damping, cocontraction levels ( explained in detail in the latter half of the chapter) and to allow direct control of servo valve currents [4]. This allows the analog controller to operate either in a stand alone mode, requiring no digital electronics, or in conjunction with the digital system. The digital system can supply desired angles, tendon tensions, cocontraction levels, positional gains, damping gains, and/or tendon tension gains. The digital system also has access to scaled analog voltages corresponding to any of the sensor signals [5].

## 3.2 CONTROL OF A SINGLE JOINT

The schematic of the control setup for a single joint is as shown in figure 3.1. The block diagram of the single joint control is as shown in figure 3.2. The analog controller is of the proportional-plus-derivative type. The analog controller has a slightly different configuration (figure 3.2) than that of the conventional proportional

plus derivative control scheme. The difference in configuration is because the movement of a joint is effected by the interaction of the two actuation systems (flexion and extension) set up in an antagonist configuration. The flexion and extension of a joint is shown in figure 3.3. A joint is defined to be flexing when the finger curls inwards. This movement is effected when the flexor tendon is pulled. A joint is defined to be extending when the the finger moves outward. This movement takes place when the extensor tendon is pulled by the extensor actuation system. The movement of a joint takes place not due to the force acting on flexor or extensor tendons only, but due, to the difference in the forces acting on the flexor and extensor tendons.

The proportional plus derivative type controller uses a series-feedback configuration as shown in the block diagram of figure 3.4 [6]. The derivative part of the controller is placed in the feedback path, while the proportional part is placed in series with the plant. The actual position, sensed by the position Hall effect sensors, is fed back as a scaled voltage between –10V and +10V. When the finger is straight out i.e. fully extended, the joint angle is zero radians, which corresponds to -10V output of the Hall effect sensor. As the finger flexes, the joint angle increases positively from 0 radians and the transducer output increases positively from –10V to +10V. This is shown in figure 3.5, where the non-zero joint angles are represented as $\theta_{xy}$ where x denotes the finger and y denotes the joint. When the finger is fully flexed, the joint angle is maximum and the transducer output is +10V (the calibration of the joint angle sensors is discussed in the next chapter). The desired position ($\theta_d$) and the actual position ($\theta_a$) in radians is converted to desired and actual position in volts by multiplying with the transducer (Hall effect sensor) constant $K_T$, as shown in figure

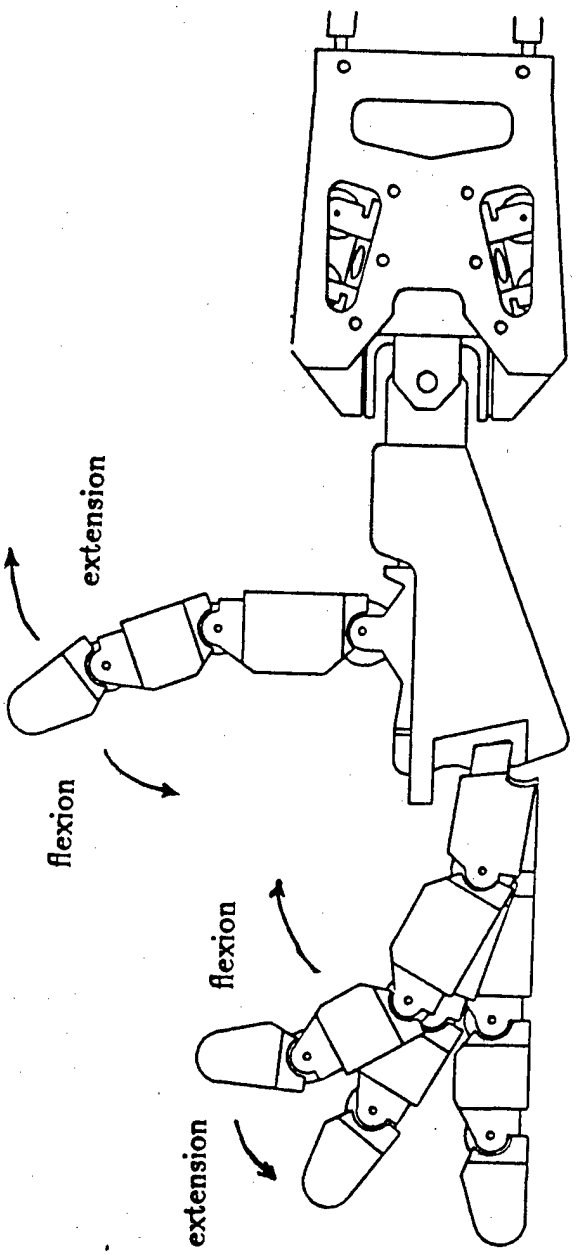



**Figure 3.3 Line drawing of the Utah/M.I.T. DH illustrating flexion and extension motion**

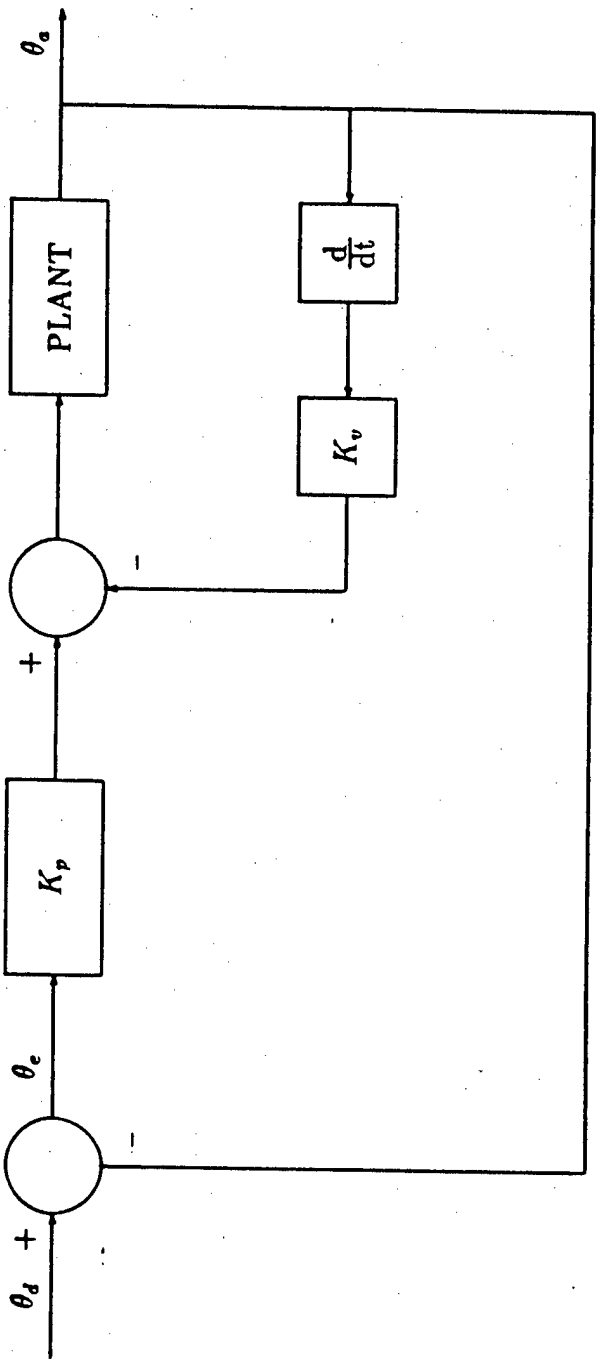



**Figure 3.4 Series-feedback configuration**

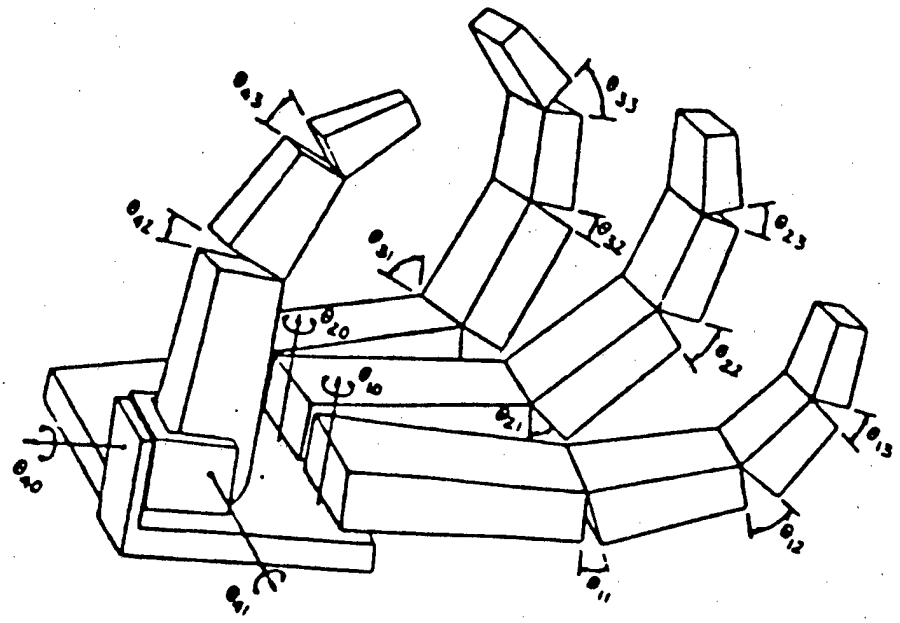



**Figure 3.5 Line drawing of the Utah/M.I.T. DH showing joint angles [7]**

3.2. This can be represented as

$$V_d = K_T\, 0_d \tag{3.1}$$

$$V_a = K_T\, 0_a \tag{3.2}$$

The error between the desired and the actual position voltage is passed through a position gain potentiometer, whose gain $K_p$ can be varied. This can be written as

$$M_p = K_p\, [V_d - V_a] \tag{3.3}$$

where

$K_p$ – spring constant or stiffness in lb-in/volt

$V_d$ – desired angular position in volts

$V_a$ – actual angular position in volts

$M_p$ – moment generated due to positional error in lb-in

The actual position voltage $V_a$ is passed through a differentiator, and then to a velocity gain control potentiometer which has a variable gain $K_v$. This can be represented as

$$M_v = K_v\, K_f\, \frac{\mathrm{d}}{\mathrm{dt}}\, (V_a) \tag{3.4}$$

$K_v$ – viscous frictional coefficient in lb-in/volt/sec.

$M_v$ – moment generated due to the actual velocity of joint in lb-in.

$K_f$ – gain factor associated with the differentiator circuit.

These moments are then passed through half-wave rectifiers whose operating characteristics are shown in figure 3.6. The $R^+$ and $R^-$ operators in figure 3.6 can be represented mathematically as

$$R^+(x) = 1/2[\, x + |x|\, ] \tag{3.5}$$

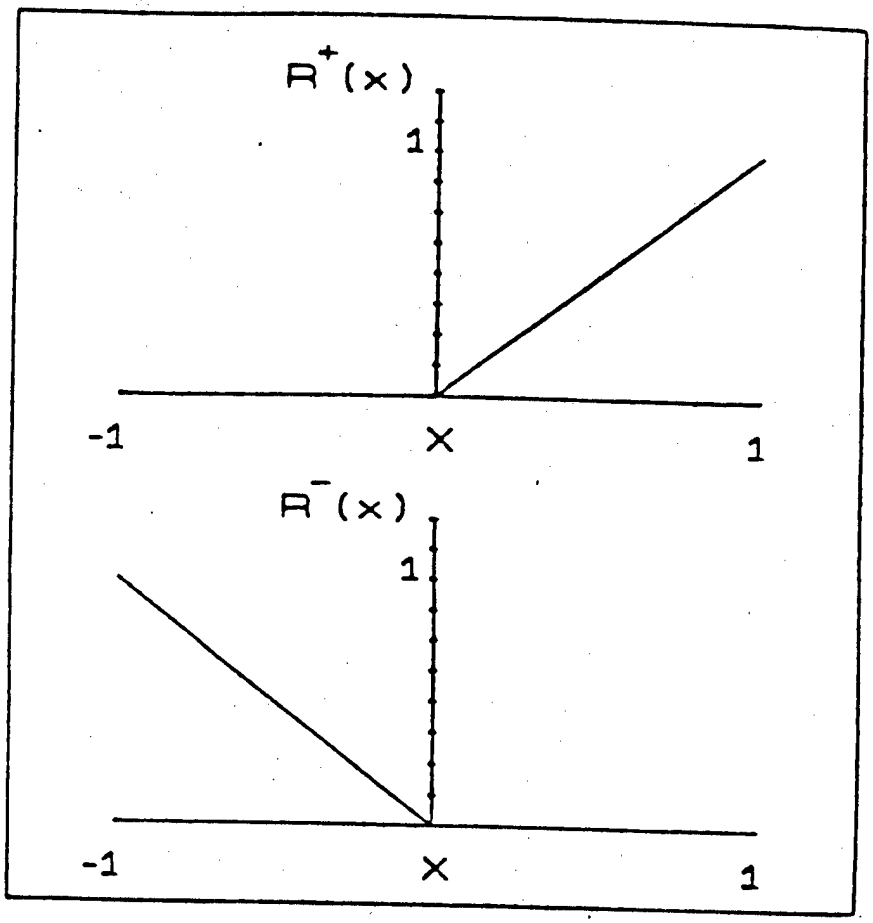



**Figure 3.6 Operating characteristics of the half-wave rectifiers [7]**

$$R^{-}(x) = 1/2[\,-x + |x|\,] \tag{3.6}$$

where x is the input to the half-wave rectifier.

The $R^{+}$ operator suppresses all negative moments, and allows only positive moments. This side is known as the flexor tendon system and the output of the rectifier is essentially the desired flexor tendon tension. The $R^{-}$ operator suppresses all positive moments and allows only negative moments, this side is known as the extensor tendon system and the output of the half-wave rectifier is essentially the desired extensor tendon tension. Thus, at any particular time, moment is being applied either to the flexion tendon system or to the extension tendon system. When moment is being applied to one system, the tendon tension in the other system is zero i.e. the tendon goes slack. This can be seen in figure 3.7, which shows the underdamped step response and the flexion and extension tendon tension of a joint.

The step input is shown in dotted lines. It is evident from figure 3.7 that when the error between the desired and actual position is positive, the flexor tendon tension exhibits a non-zero value and the extensor tendon tension is zero. When the error is negative, the extensor tendon tension is non-zero and the flexor tendon is at zero tension state, i.e. the tendon goes slack. Also spikes appear when tendon tensions change from zero to a non-zero value. This phenomenon is known as **snapping of tendons**. Hence, to avoid the zero tension state when no moment is applied to the system, a constant bias term is added on both the flexion and extension system. This bias term is known as **cocontraction**. Biologists refer to this term in humans as co-activation. The effect of cocontraction can be seen in figure 3.8. The tendons

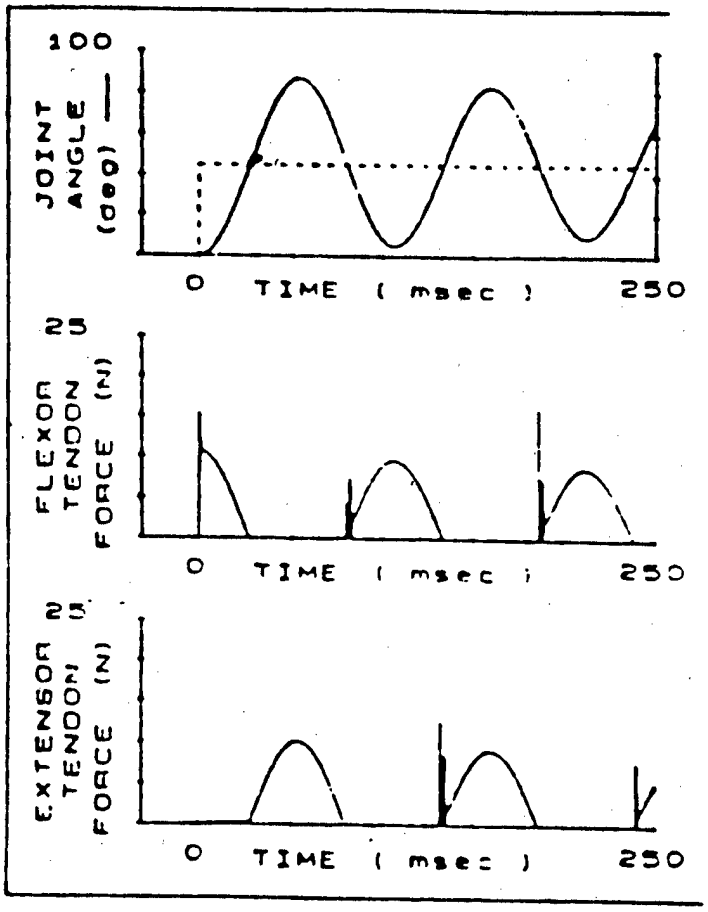



**Figure 3.7 Response of joint with no cocontraction input [7]**

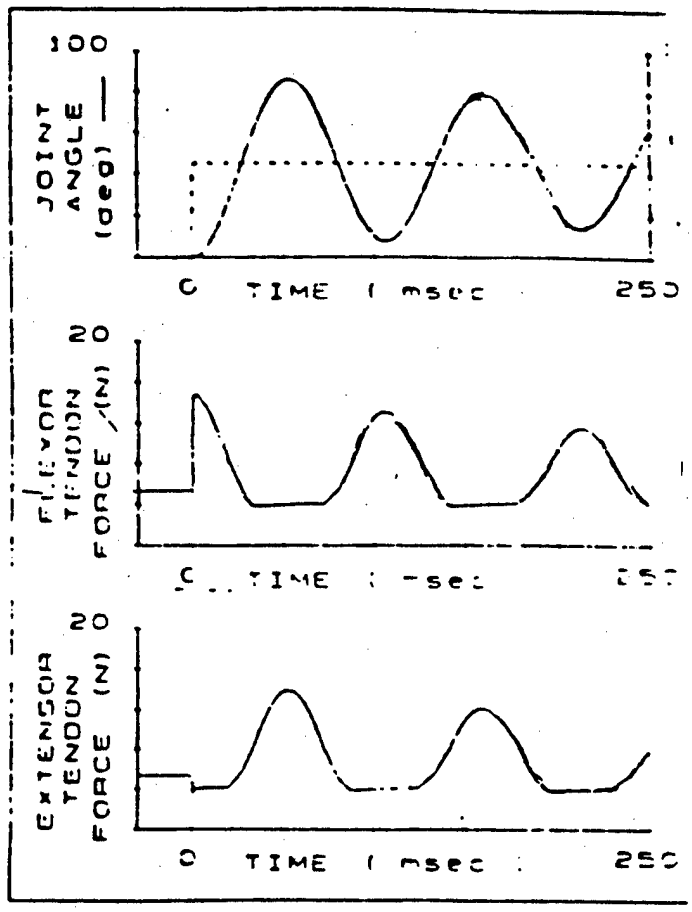



**Figure 3.8 Response of joint with cocontraction input [7]**

never go to zero tension state, in other words, the tendons never go slack. This results in the elimination of the snapping action. The tendon management discussed above can be mathematically represented as

$$\begin{pmatrix} T_{fd} \\ T_{ed} \end{pmatrix} = \begin{pmatrix} R^{+} \\ R^{-} \end{pmatrix} M_p + \begin{pmatrix} R^{+} \\ R^{-} \end{pmatrix} M_v + \begin{pmatrix} B_f \\ B_e \end{pmatrix} \tag{3.7}$$

where

$T_{fd}$ – desired flexion tendon tension

$T_{ed}$ – desired extension tendon tension

$B_f$ – flexion bias (cocontraction input to the flexion system)

$B_e$ – extension bias (cocontraction input to the extension system)

The analog control system uses a force feedback loop on each tendon to regulate the moments applied to the finger joints by the pneumatic actuators. This includes linearizing the actuator and removing the effects of stiction in the tendon pulley system. The actual tendon tension is detected by the tendon tension sensors located at the wrist joint. The gains $K_e$ and $K_f$ determine the stiffness of the force loop. The desired torques or moments obtained to produce desired tendon tensions are signal-conditioned and a proportional current is applied to the valve. The valve outputs an equivalent pressure which drives the actuation system. The actuator pulls on the tendon which results in the movement of the joint.

## 3.3 DEXTEROUS HAND CONTROL DEVELOPMENT SYSTEM

The current Utah/M.I.T. Dexterous Hand (DH) control development system hardware is shown in figure 3.9. The control development system was developed for the following reasons :

1. To perform different tests on the DH.
2. To conduct frequency response analysis in order to identify the transfer functions of different joints and that of coupling between different joints of the Utah/M.I.T. DH.
3. To set the various parameters of the analog servo controller to different values.
4. To slave the DH to an exo-skeleton master worn by a human operator.

The major components of the DH control system hardware are described below.

### 3.3.1 IBM PC-AT

THE IBM PC-AT acts as a host for software development to slave the DH to an exo-skeleton master worn by a human operator and for system operation. It has 2 Mbytes of RAM, a 30 Mbyte hard disk, dual floppy disk drives, a Hercules monochrome graphics adaptor, 80287 math chip, 2 serial ports, 3 printer ports, and MS-DOS v3.10. Two printer ports are used for the printer and plotter (figure 3.9). A serial port is used to link the PC to the VMEbus system. This enables different commands to be given to the dexterous hand analog servo controller from the PC.

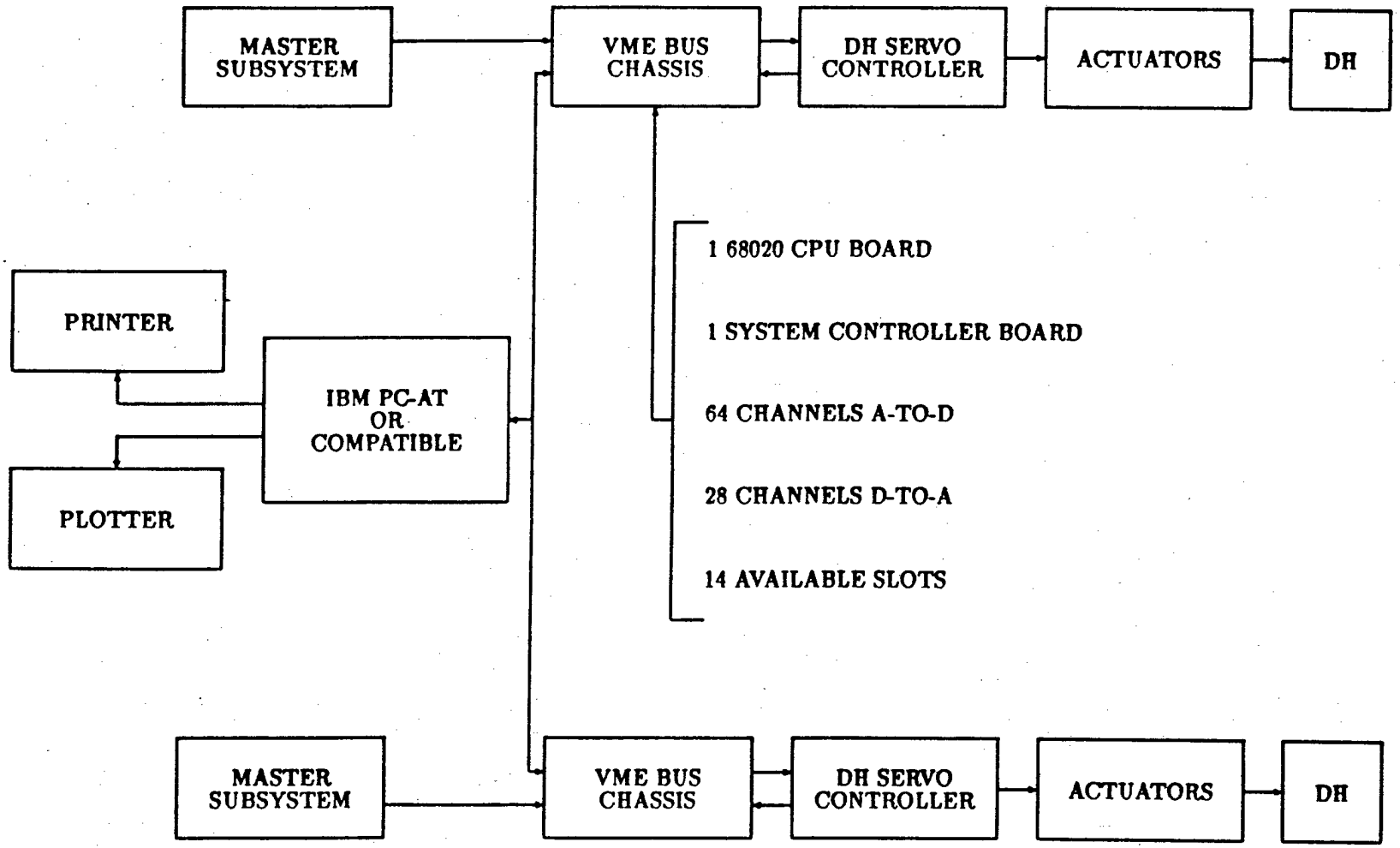



**Figure 3.9 CONTROL SYSTEM BLOCK DIAGRAM FOR THE UTAH/M.I.T. DEXTEROUS LEFT AND RIGHT HAND SYSTEM**

Available languages for developing PC applications include: Turbo C, Fortran, Turbo Pascal, and assembly. All DH software has been developed using Turbo C and MC80286 Assembly (80286/80287). A library of routines for creating data files in a standard format and accessing them has been developed. Library routines for downloading and uploading raw data files plus a terminal emulator with download support are available. Also spoolers and queue managers for plots and data tables have been developed. A library of routines for the I/O devices (serial ports, analog to digital converters, digital to analog converters, a Random Access Memory file and timer interrupt services) is available.

### 3.3.2 VMEBUS CHASSIS

The VMEbus chassis contains a system controller board (Ironics IV-3273) with 2 serial ports, 2 parallel ports, a clock/calendar, and an SCSI interface. The console serial port is used to link the VMEbus system to the PC host.

There is one CPU board (Ironics IV-3201) with a 16 MHz 68020 processor and a 68881 math co-processor. It has 1 Mbyte of dual-ported RAM, an EPROM monitor (IMON 3201), and a timer/counter chip. The monitor comes up on the console serial port and can be used directly with a terminal emulator program in the PC host.

There are two A/D (analog to digital) converter boards (Data Translation DT-1401). Each board has 32 12-bit A/D channels and 2 12-bit D/A channels. To match the Sarcos controller, the analog ranges are set from -10 to +10 volts full scale (giving a resolution of 4.88mV/count).

There are three D/A converter boards (Data Translation DT-1406-8V). Each

board has 8 12-bit D/A channels. As with the A/D boards the analog ranges are set from −10 to +10 volts full scale. The D/A converters have a maximum 1/2 Least Significant Byte accuracy, settling time (for a full scale step) of 8$\mu$sec. and a 5 V/$\mu$sec. slew rate. Totally, there are 64 ADC (analog-to-digital channels) and 28 DAC (digital-to-analog channels) available.

### 3.3.3 SARCOS DH CONTROLLERS

The two Sarcos DH controllers for the left and right dexterous hands have several modes of computer controlled operation ranging from position servo to force servo to direct valve control. All I/O (input/output) with the controllers is done through analog signals (−10V to +10V). Each controller can accept up to 112 command signals output from a computer.

Since only a very few of the many possible connection patterns will be needed at any time, different cables were made to connect different parts of the controller to the ADC/DAC. Currently three cables are available: the first connects all 64 data signals from the controller to the 64 ADC. The second connects the 28 DAC to all the possible command signals for one finger of a DH. The third connects 16 DAC to the 16 command positions of a controller plus 1 DAC to all 16 position gains, 1 DAC to all 16 velocity gains, 1 DAC to all 32 tension gains and 1 DAC to all 32 tension biases.

### 3.3.4 DATA ACQUISITION SOFTWARE

In addition to the hardware and software development tools, a simple data acquisition system is available. The Data Acquisition and Waveform Generator System (DAWGS) allows 27 DAC outputs to be set to constant values and one DAC output to generate a sine or square wave function. While the output function is occurring, up to 20 ADC inputs may be recorded. Data loops can be recorded at 250 $\mu$sec. (for 1 or 2 channels), at 500 $\mu$sec. (for 9 or less channels) or at 1000$\mu$sec. intervals. Approximately 400,000 ADC samples could be recorded in a single data set. Recorded data may be tabulated or plotted versus time.

The information on the DII control development system has been provided by Systems Research Laboratory, who developed the system. The calibration of the various gain blocks in the analog servo controller and those of the joint angle and position sensors is discussed in the next chapter.

# IV . CALIBRATION

Many abbreviations are used in this chapter. The explanation of these abbreviations are given in Appendix A.

## 4.1 NEED FOR CALIBRATION

As mentioned in the previous chapter, each joint controller contains the following gain blocks (figure 4.1) :

1. Position gain block (GP)
2. Velocity gain block (GV)
3. Flexion gain block (GF)
4. Extension gain block (GE)

These gain blocks are built with the help of operational amplifiers and potentiometers. The potentiometers act as voltage dividers. One end of the potentiometer is connected to −10V while the other end is connected to +10V. Depending on the voltage setting, the potentiometer has a gain value varying from 0 to 10. Hence, the potentiometers need to be calibrated to determine the gains produced corresponding to their voltage settings. Also, each joint has a position Hall effect sensor and a tendon tension sensor. The Hall effect sensor outputs a voltage corresponding to

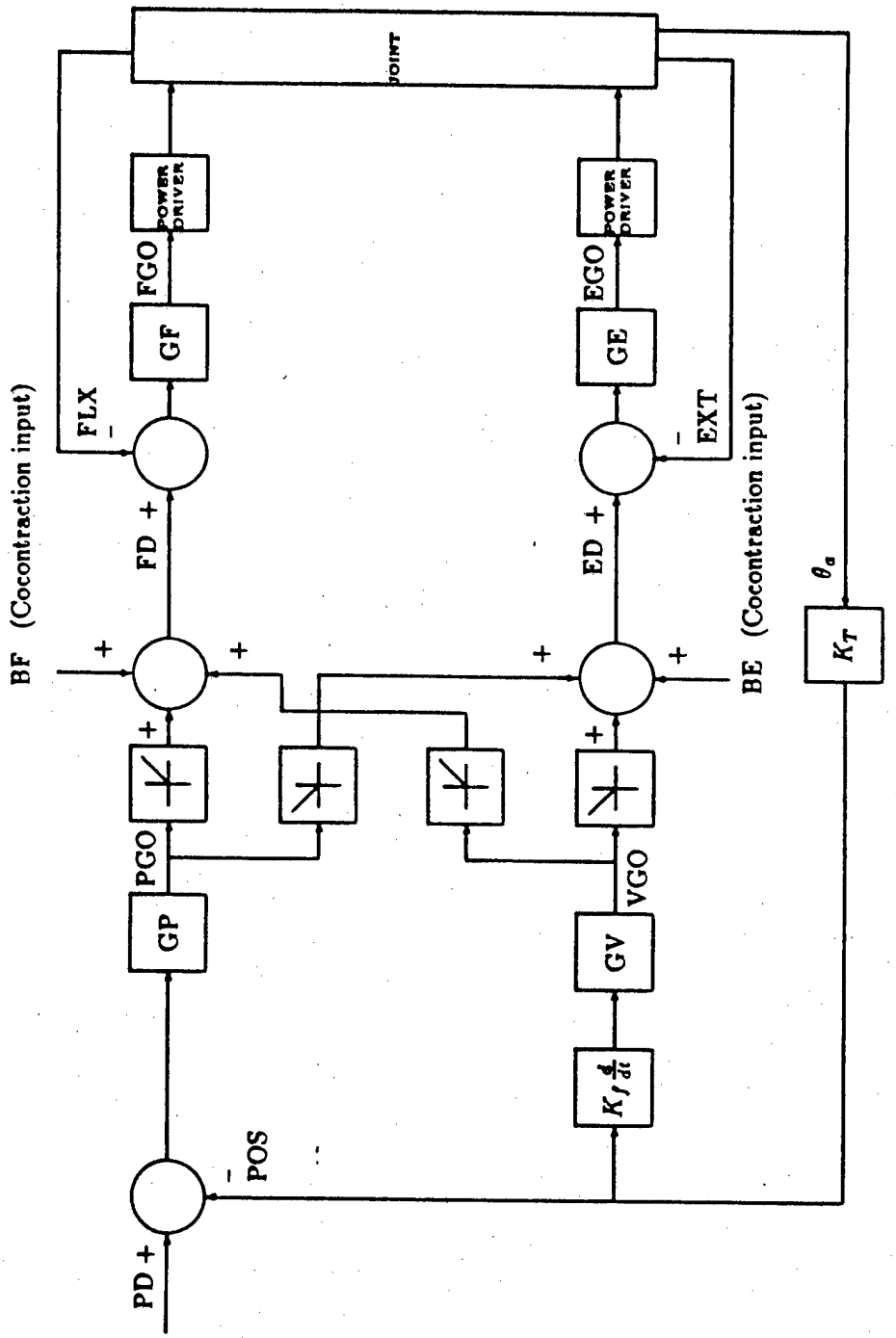



**Figure 4.1 Block diagram of the position control loop of a single joint showing the various blocks to be calibrated**

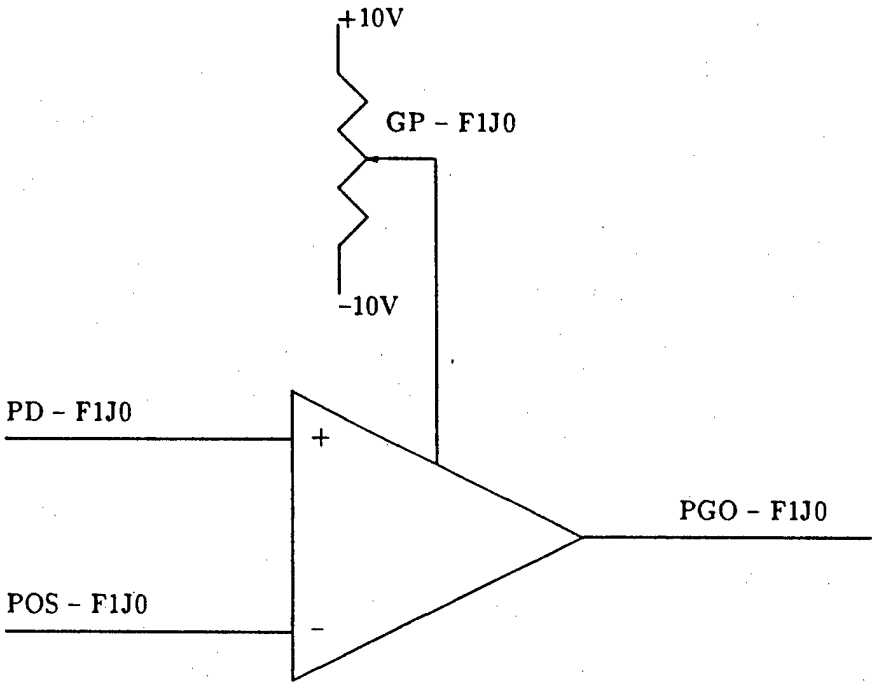



**Figure 4.2 Position gain block**

the joint angle, with a proportionality constant $K_T$ as shown in figure 4.1. The relationship between the voltage output of the sensor and the joint angle needs to be determined. Similarly, the tendon tension sensor outputs a voltage proportional to the tendon tension measured. This sensor needs to be calibrated to determine the relationship between the voltage produced and the tendon tension.

## 4.2 CALIBRATION OF THE GAIN BLOCKS

The position, extension and flexion gain blocks incorporate both the summing block and the gain block in one operational amplifier unit. The position gain block has two inputs namely desired position (PD) and actual position (POS) (figure 4.2.).

The error between the two is multiplied by a gain factor. The gain factor was calculated by dividing the output (PGO) by the difference between the two inputs of the operational amplifier (PD and POS), obtained for different values of gain voltage settings (GP) varying from -10V to +10V. The gain factors of the position gain blocks of F1J0 and F1J1 were plotted against the corresponding potentiometer settings (in volts) for two different values of desired and actual input voltages (figure 4.3). The two slopes differed slightly from each other only for the high gains, while for the low gains, they were very much the same. The calculations are shown in Appendix B. Since the differences were small for the high gains, a straight line was fitted through these points, which took the average of these values. Thus the relationships between the gain setting in volts and the gain factor for F1J0 and F1J1 were determined. The calibration plots and tables of F1J2 and F1J3 are shown in Appendix B.

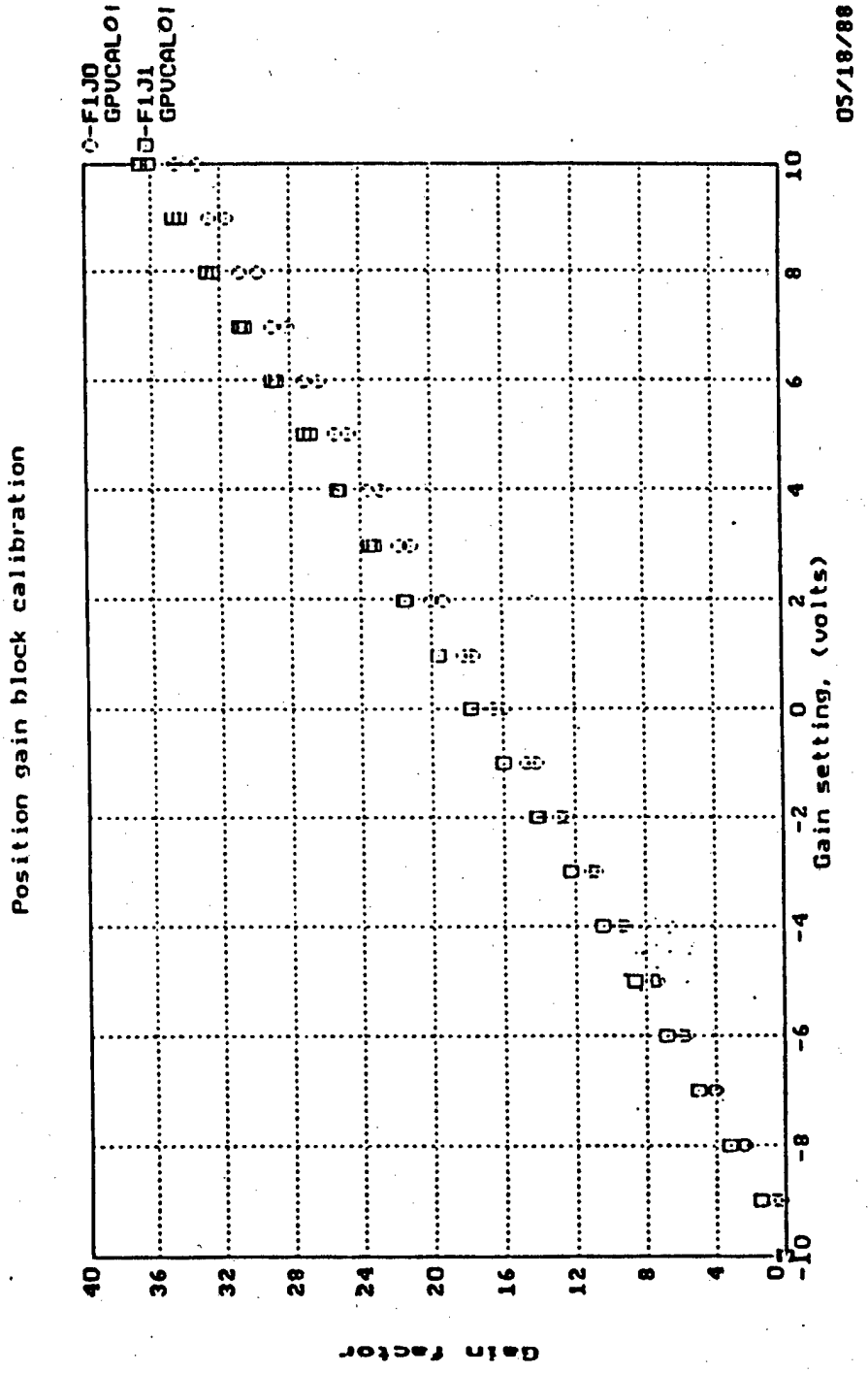



Figure 4.3 Position gain block calibration

The flexion (extension) gain block calibration was carried out in the same manner. The difference between the desired input FD (ED) and the actual input FLX (EXT) is calculated as the input error. The gain factor of the flexion (extension) gain block was obtained by dividing the gained output FGO (EGO) by the input error between FD (ED) and FLX (EXT) for different values of the gain voltage settings GF (GE) varying from -10V to +10V. The block diagrams of the flexion and extension gain blocks is given in figures 4.4 and 4.5 respectively. The gain settings in volts of the flexion (extension) blocks of F1J0 and F1J1 were plotted against the gain factors as shown in figure 4.6 (figure 4.7). Thus the relationship between the gain settings and gain factors for the above-mentioned gain blocks were determined. The calibration tables and plots of F1J1, F1J2 and F1J3 are given in Appendix B.

The calibration of the velocity gain block (figure 4.8) was carried out in a slightly different manner, because there is only one input ( actual velocity (VEL)). The gain voltage (GV) was varied from -10V to +10V, and the gained output voltage was measured. This procedure was repeated for different values of the input (VEL). The gain factor was obtained by dividing the output by the input error (which is the same as the actual input). The gain setting in volts was plotted against the gain factor as shown in figure 4.9, which shows plots for F1J0 and F1J1. The gain factor was the same for both values of the input. The calibration tables and graphs of the velocity gain blocks of F1J1, F1J2 and F1J3 are given in Appendix B.

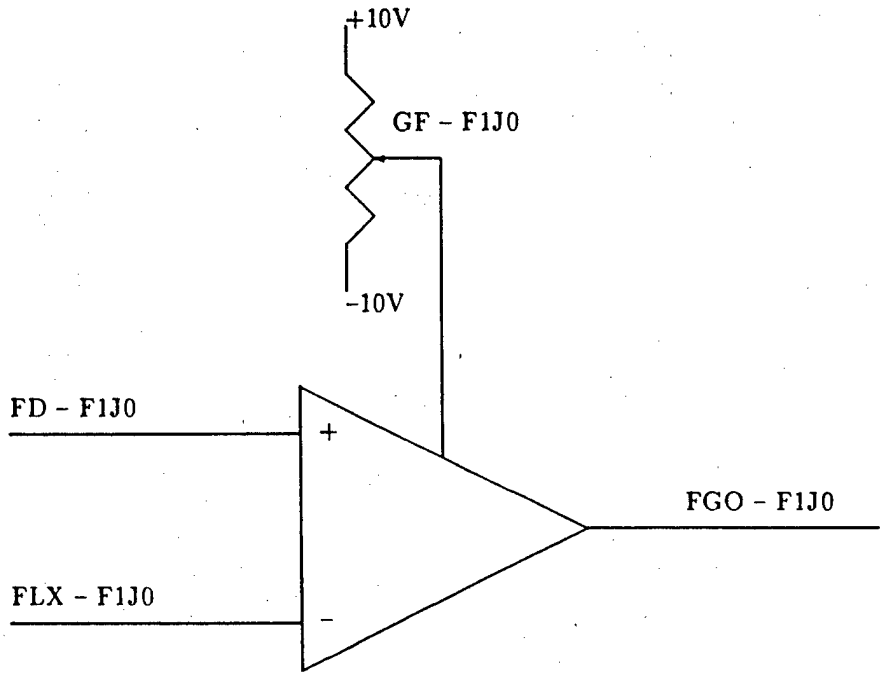



**Figure 4.4 Flexion gain block**

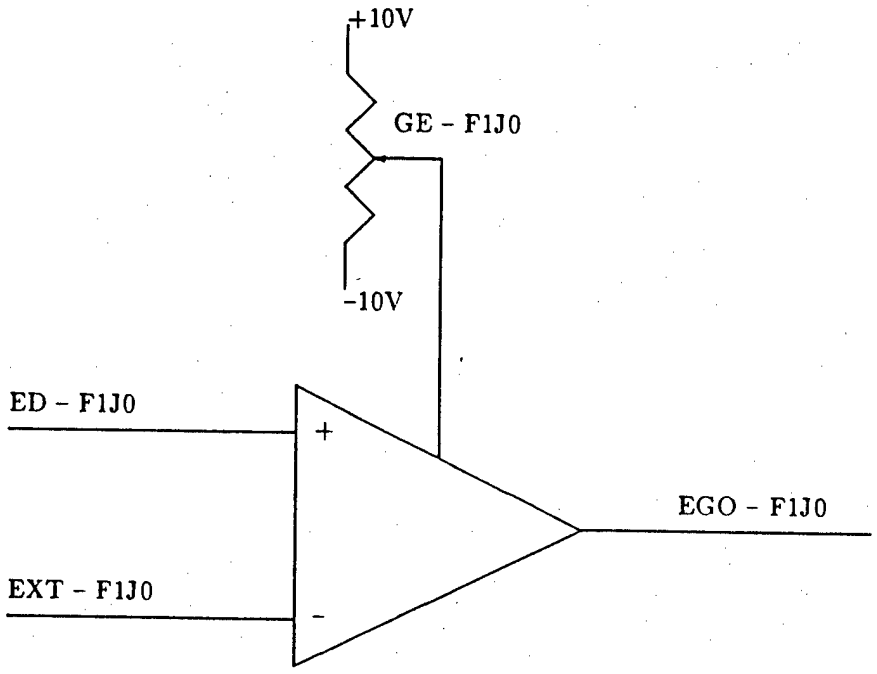



**Figure 4.5 Extension gain block**

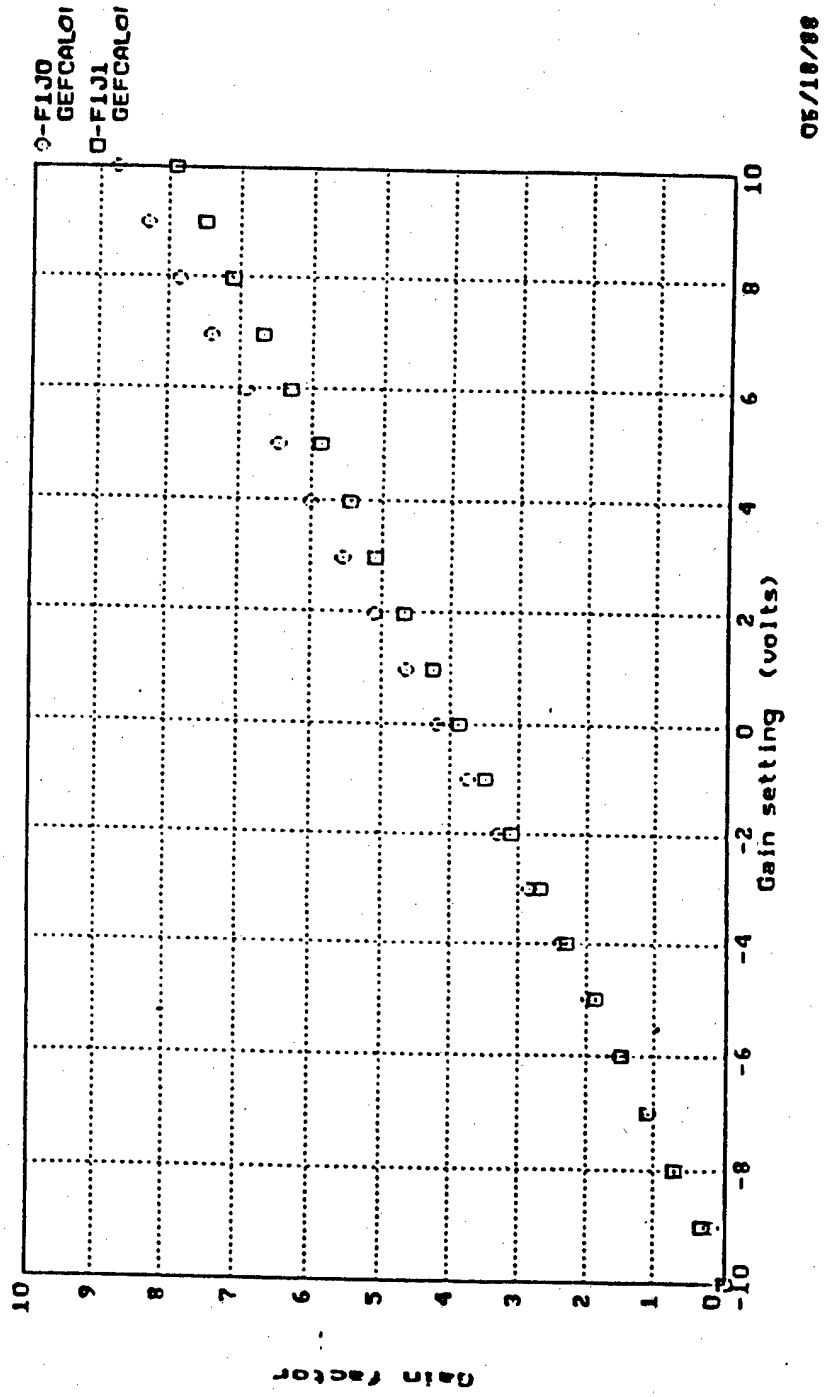



Figure 4.6 Flexion gain block calibration

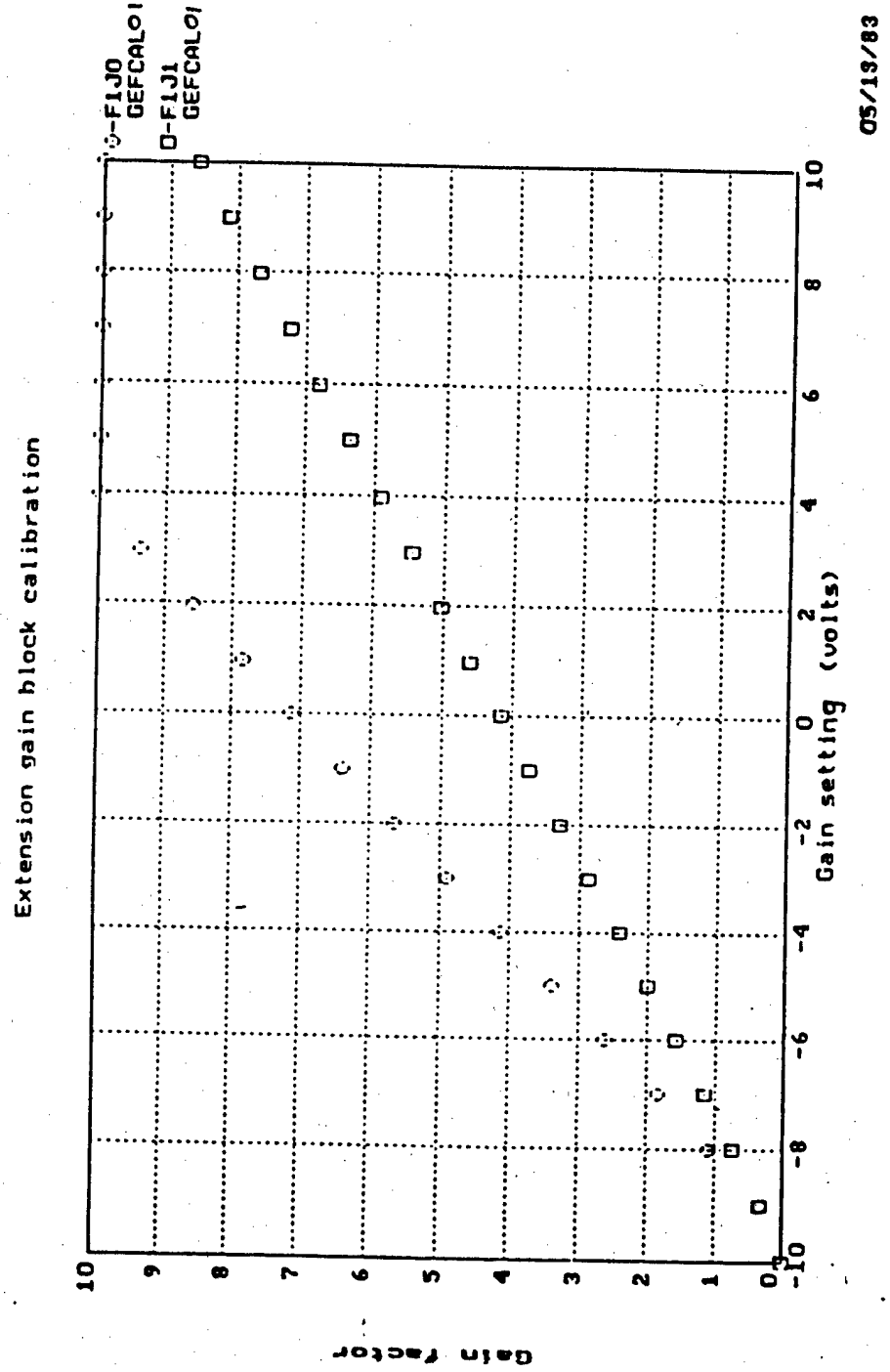



Figure 4.7 Extension gain block calibration

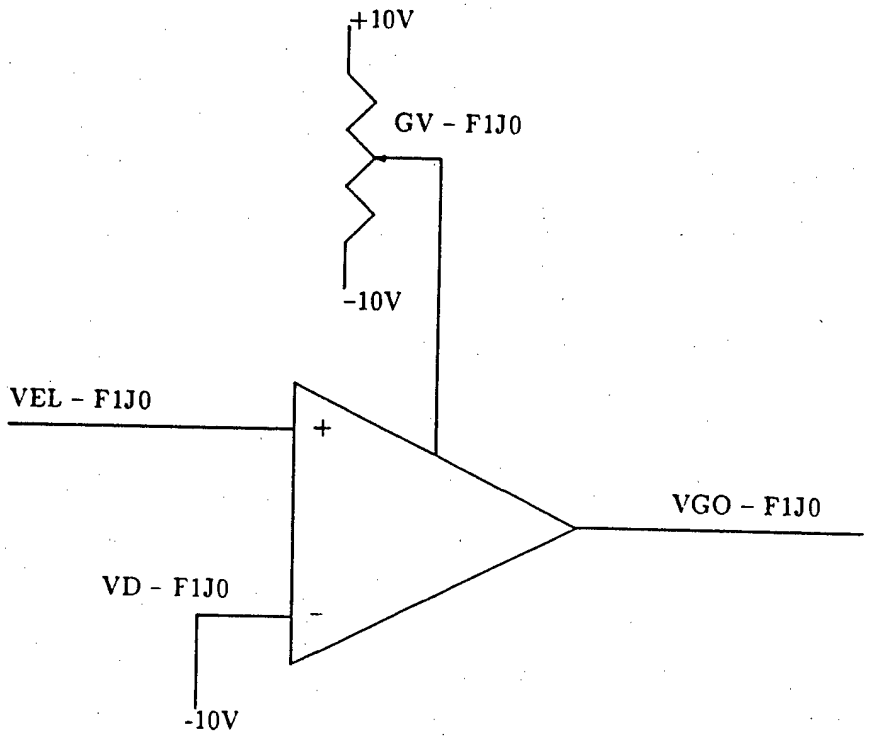



**Figure 4.8 Velocity gain block**

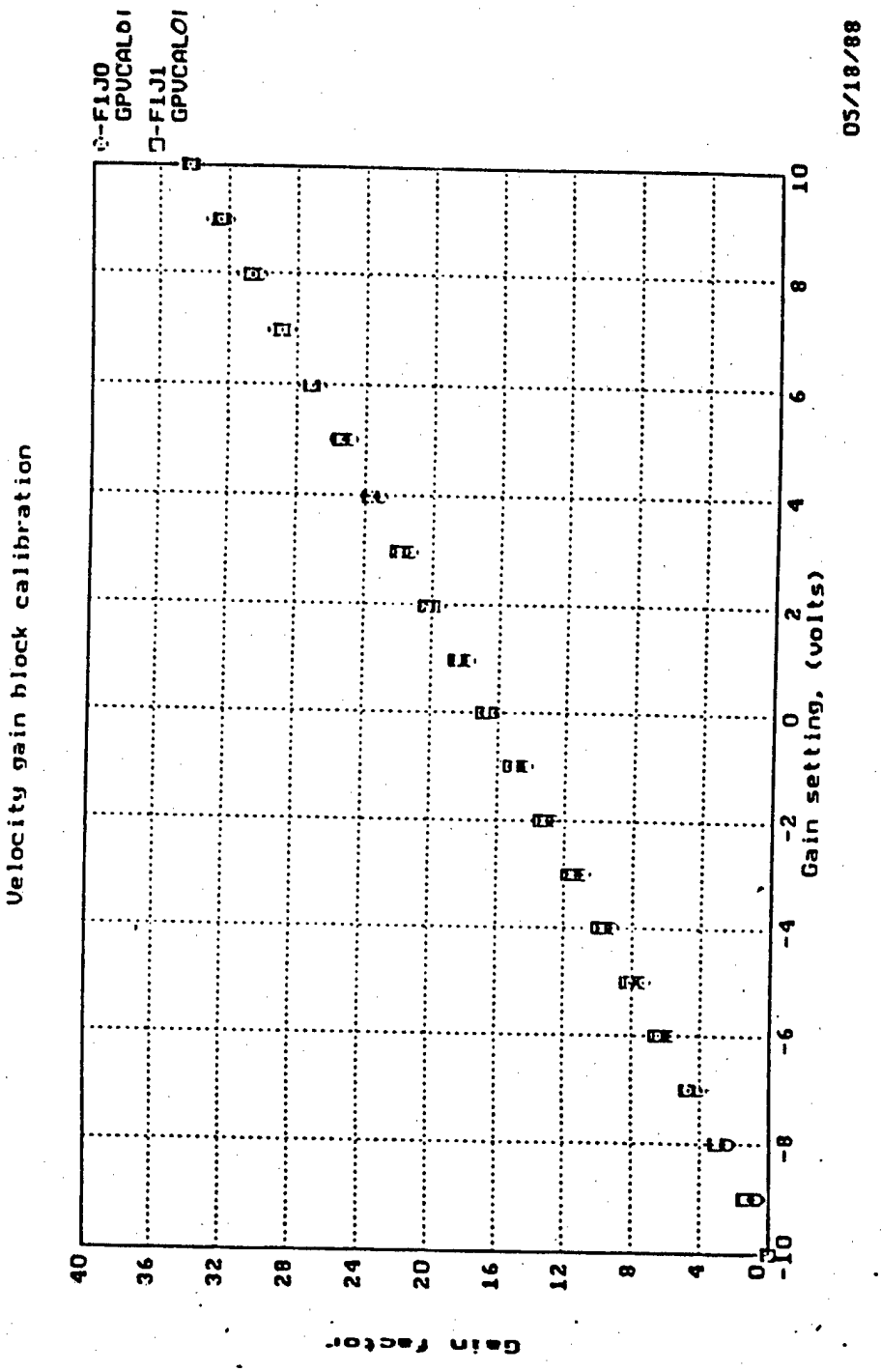



Figure 4.9 Velocity gain block calibration

## 4.3 CALIBRATION OF THE JOINT ANGLE SENSORS

The joint angle sensors are the Hall effect sensors, located at the side of each joint. The sensors produce a voltage proportional to the angular displacement. The sensors were calibrated with the help of an 'E113 Series Miniature Incremental Rotary Optical Encoder' (figure 4.10).

The E113 encoder provides high precision angular position and velocity signals. The design consists of a patented glass disc and hub assembly, focused infrared light source, a monolithic photodetector array and signal conditioning electronics. For a 360° rotation, the encoder outputs 1024 pulses. By mounting the optical encoder on the side of the joint, the rotation of the joint can be determined by counting the number of pulses outputted by the optical encoder. This gives an accurate measure of the joint angle displacement. The joint angle is calculated using the formula

$$\theta = \frac{x}{360} \times 1024 \text{ degrees} \tag{4.1}$$

where

$\theta$ – joint angle displacement

x – number of pulses outputted.

After pre-calibrating the optical encoder as explained, the encoder was mounted on the side of joint 1 of finger 1. The joint was moved from 0 degrees position to full scale deflection and moved back to 0 degrees. The output voltage of the Hall effect sensor was passed through an A/D converter and read off as counts. This was compared to the pre-calibrated, standardized output of the E113 optical encoder. The counts of the A/D converter (POS-F1J1) was plotted against the pre-calibrated

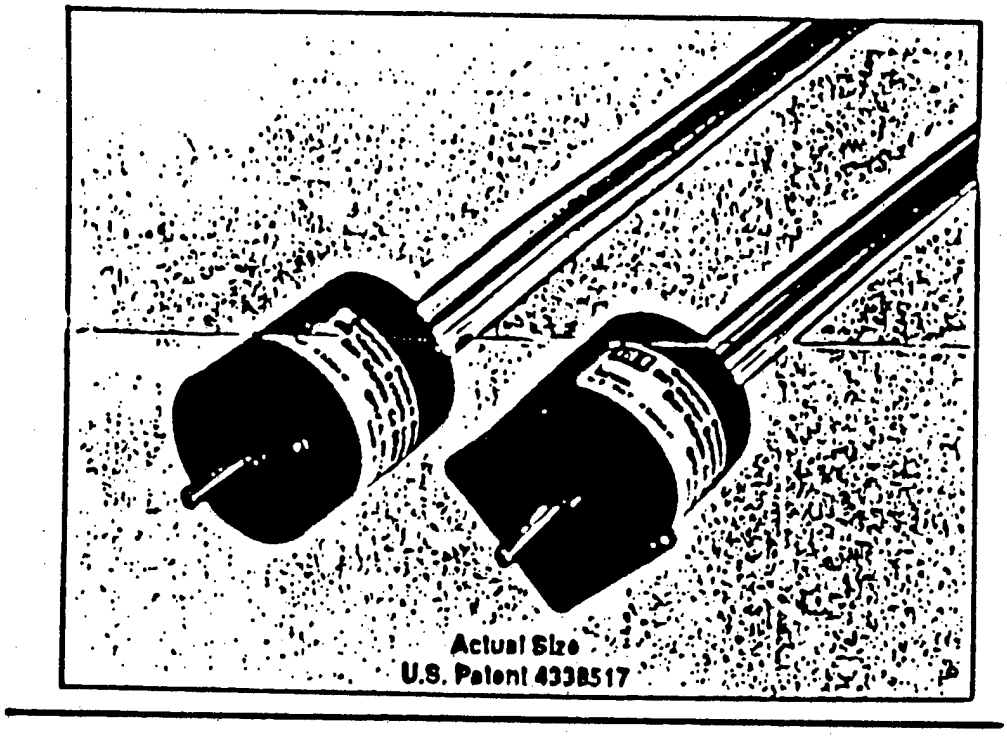



**Figure 4.10 E113 Seies Miniature Incremental Rotary Optical Encoder**

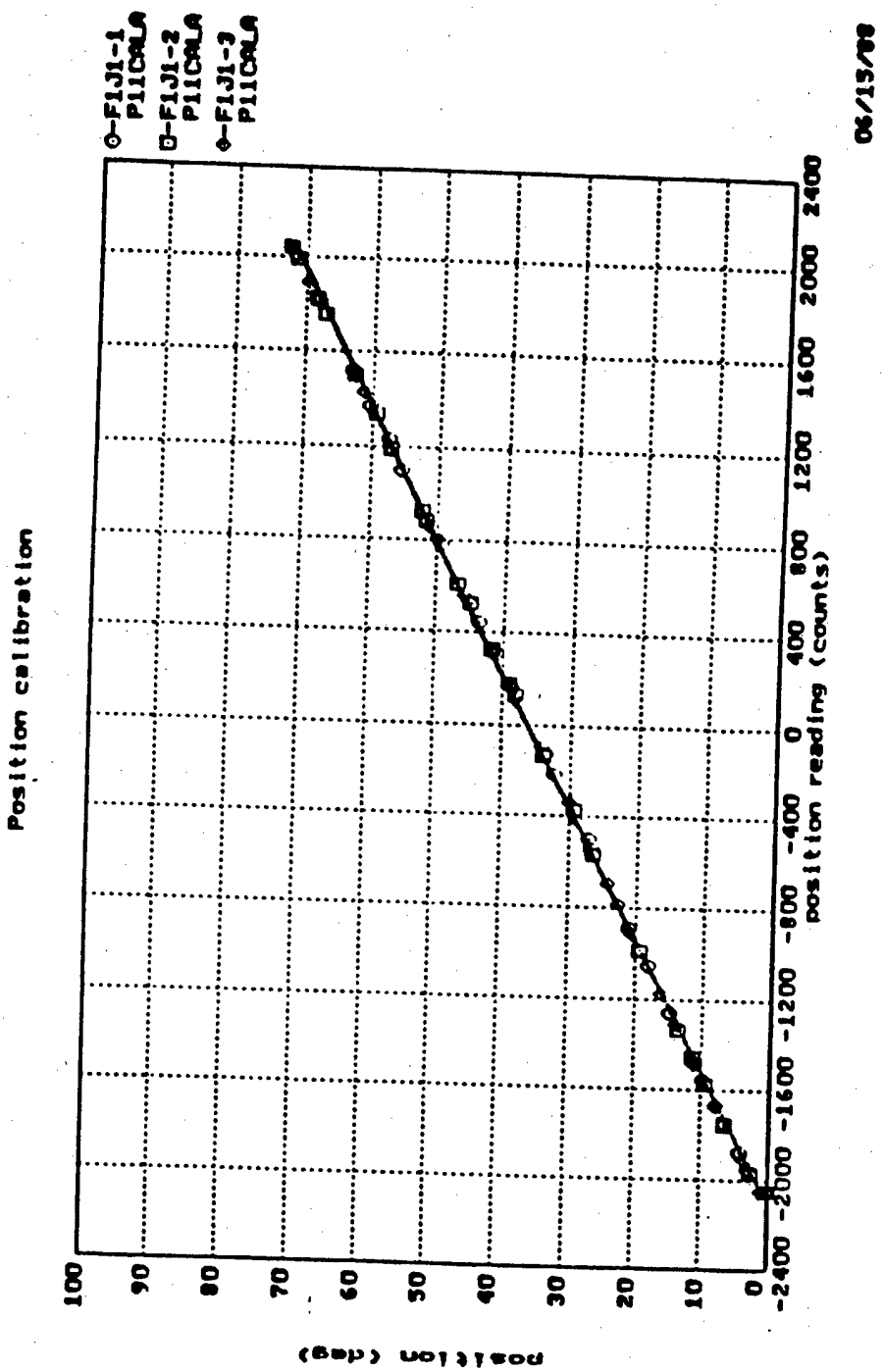



Figure 4.11 Position sensor calibration

output (in degrees) of the encoder (ENC-F1J1), as shown in figure 4.11, and a straight line was fitted through these points. This was done to determine the equation that would convert the position reading (in counts) of the Hall effect sensors obtained from the A/D converters into position reading (in degrees). This procedure was repeated two more times to test the joint angle sensor for consistency . The values obtained were tabulated as shown in Appendix B. It can be seen from figure 4.11 that the Hall effect sensor produced consistent results all the three times. The equation of conversion of the position reading (counts) to the position in terms of degrees is given in Table B17 in Appendix B. The Hall effect sensors were found to be linear with an error of $\pm$ 0.5 degrees. The calibration plots and tabulations of the position Hall effect sensor of F1J2 and F1J3 are shown in Appendix B. Joint 0 of all the four fingers could not be calibrated in the same manner, because it was impossible to mount the encoder. So, an approximate measure of conversion obtained from the calibration results of the other joints was used since the Hall effect sensors used were of the same kind.

## 4.4 CALIBRATION OF THE TENDON TENSION SENSORS

The tendon tension sensors were calibrated using a load cell. A load cell is a transducer device that produces an output that is proportional to an applied force. The load cell which was used for the calibration is of the strain-gage type. The load cell uses strain-gages to sense the force or tendon tension and output a proportional voltage. Initially, the load cell was calibrated by hanging known weights and noting

the voltage output (in counts) which was obtained by passing the output of the load cell through an A/D converter. Once the load cell was calibrated, the flexor tendon of F1J0 was disconnected from the actuator box and connected to a rigid assembly, to which was attached the load cell. The tendon was tightened slowly which increased the tension from 0 lbf to 10 lbf. The tendon was then loosened which decreased the tendon tension to 0 lbf, and the output of the tendon tension sensors (in counts) was read from the A/D converters. These values were compared with the pre-calibrated output of the load cell. The results are tabulated as shown in Appendix B. The tendon tension sensor readings (in counts) were plotted against the pre-calibrated output (in lbf) of the load cell (figure 4.12). A straight line fit was done on these points and the equation of conversion was determined (given in Table B27 in Appendix B). This procedure was repeated for a tension range of 0-20-0 lbf and 0-30-0 lbf. The plots of all the three ranges are shown in figure 4.12. It can be seen from these figures that there is hysteresis, and that the hysteresis effect increased with the increase in the range of measurement, i.e. the hysteresis effect was more for the 0-30-0 lbf range of measurement than for the 0-20-0 lbf range. This effect might have been due to the hysteresis effect of the pulleys through which the tendons are routed. The calibration of the flexion and extension tendon tension sensors of all the other joints was conducted in the same manner. The tabulation and plots of the results are given in Appendix B.

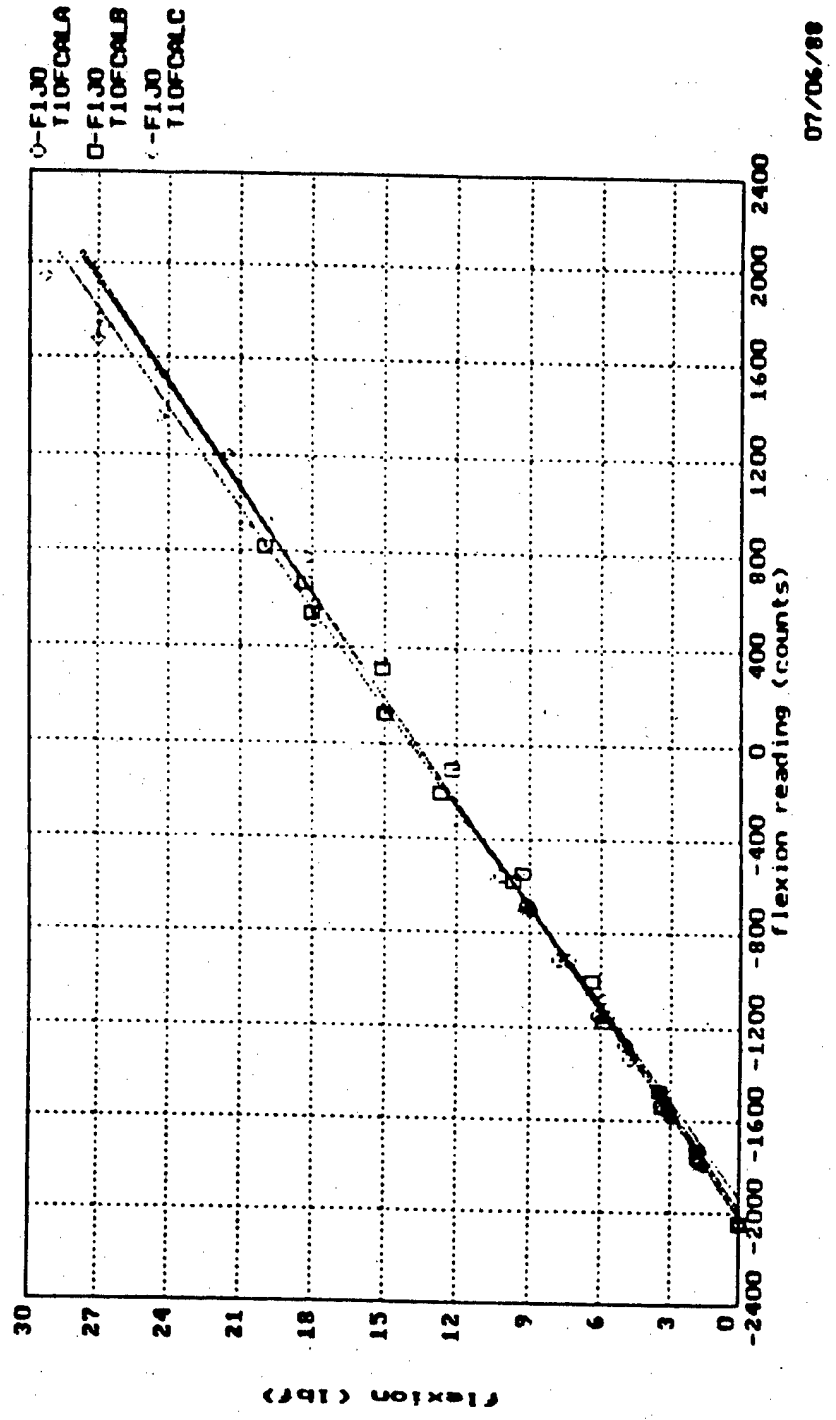



**Figure 4.12 Flexion tendon tension sensor calibration**

## 4.5 DETERMINATION OF THE BIAS SETTINGS

As mentioned in the previous chapter, a bias voltage known as cocontraction needed for both the flexor and extensor tendon tension to prevent either of the tendons from going to a zero tension state. When there is no cocontraction input, the tendons go to zero tension state or become slack. When they are brought back to a tensioned state, there is a sudden snap which shows up as spikes in the tendon tension plot, are undesirable.

The procedure followed for determining the bias setting is as follows. The position loop and the force loop (figure 4.1) were closed. The extensor and flexor blocks were adjusted to have a flexor and extensor gain of 4. This was done to ensure that the force loop remains stiff. Figures 4.13 and 4.14 show the plots obtained for the bias determination of F1J3. The values to which the various other parameters were set to are shown in figures 4.13 and 4.14. The biases $B_e$ – F1J3 and $B_f$ – F1J3 were initially set at -10V. A sine wave of frequency 5 hertz was applied to the input to the servo loop (PD-F1J3). The actual position POS-F1J3 , flexion tendon tension FLX-F1J3 and extension tendon tension EXT-F1J3 were recorded. The biases $B_e$ and $B_f$ were increased until the spikes in the plots of FLX-F1J3 and EXT-F1J3 disappeared. Figure 4.13 shows the position and tendon tension plots of F1J3 for a bias of -5.2 volts. The spikes can be seen in the plot of the output of the flexion tendon tension sensor FLX-F1J3 when the tendons reach 0 tension state (–10V). Figure 4.14 shows the plots for a tension of –3.6 volts. It can be seen that the spikes have disappeared from FLX-F1J3, and the tendon tensions never reach 0 tension state (–10V). The

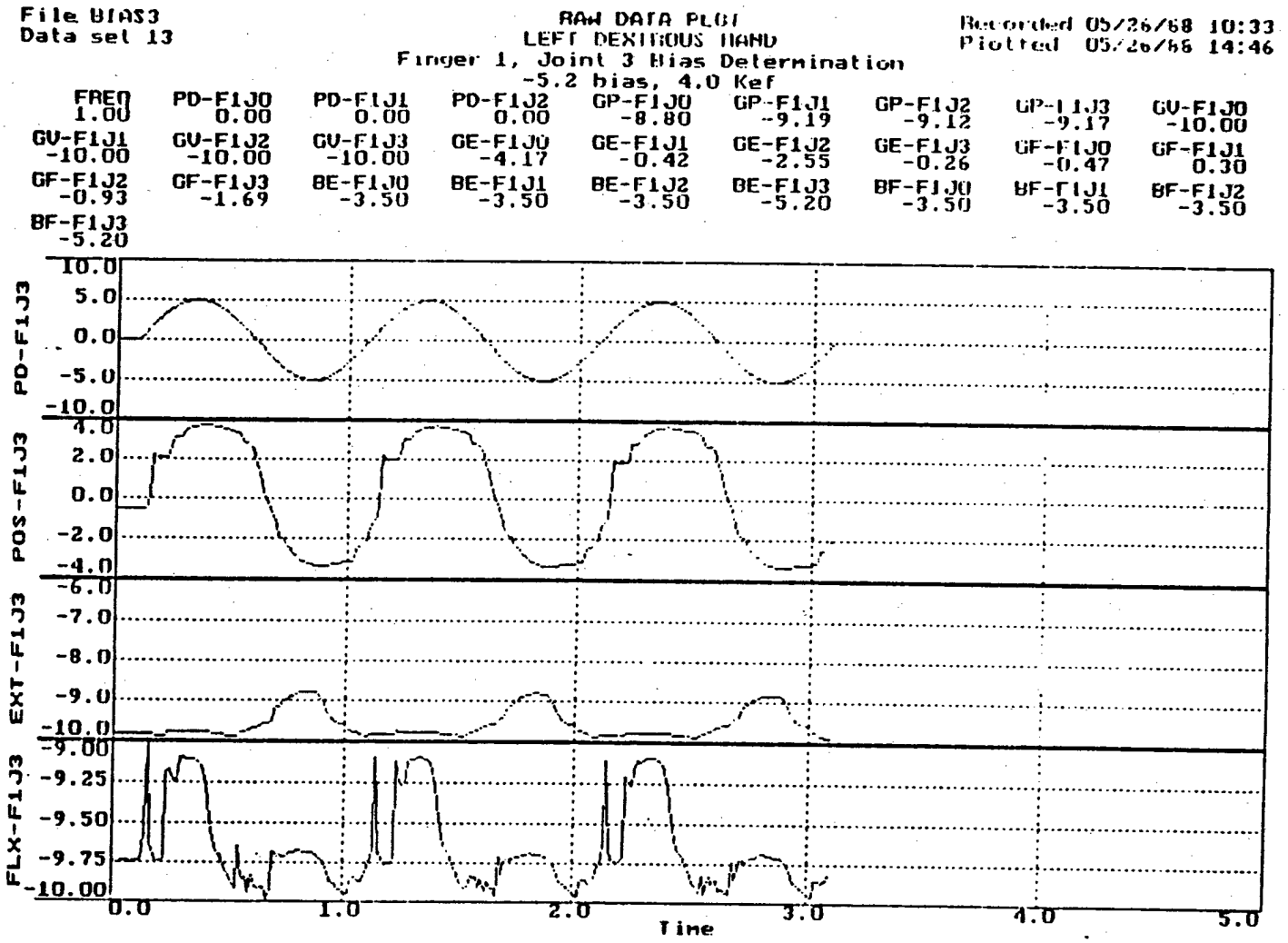



**Figure 4.13 Response of flexion and extension tendons without cocontraction input**

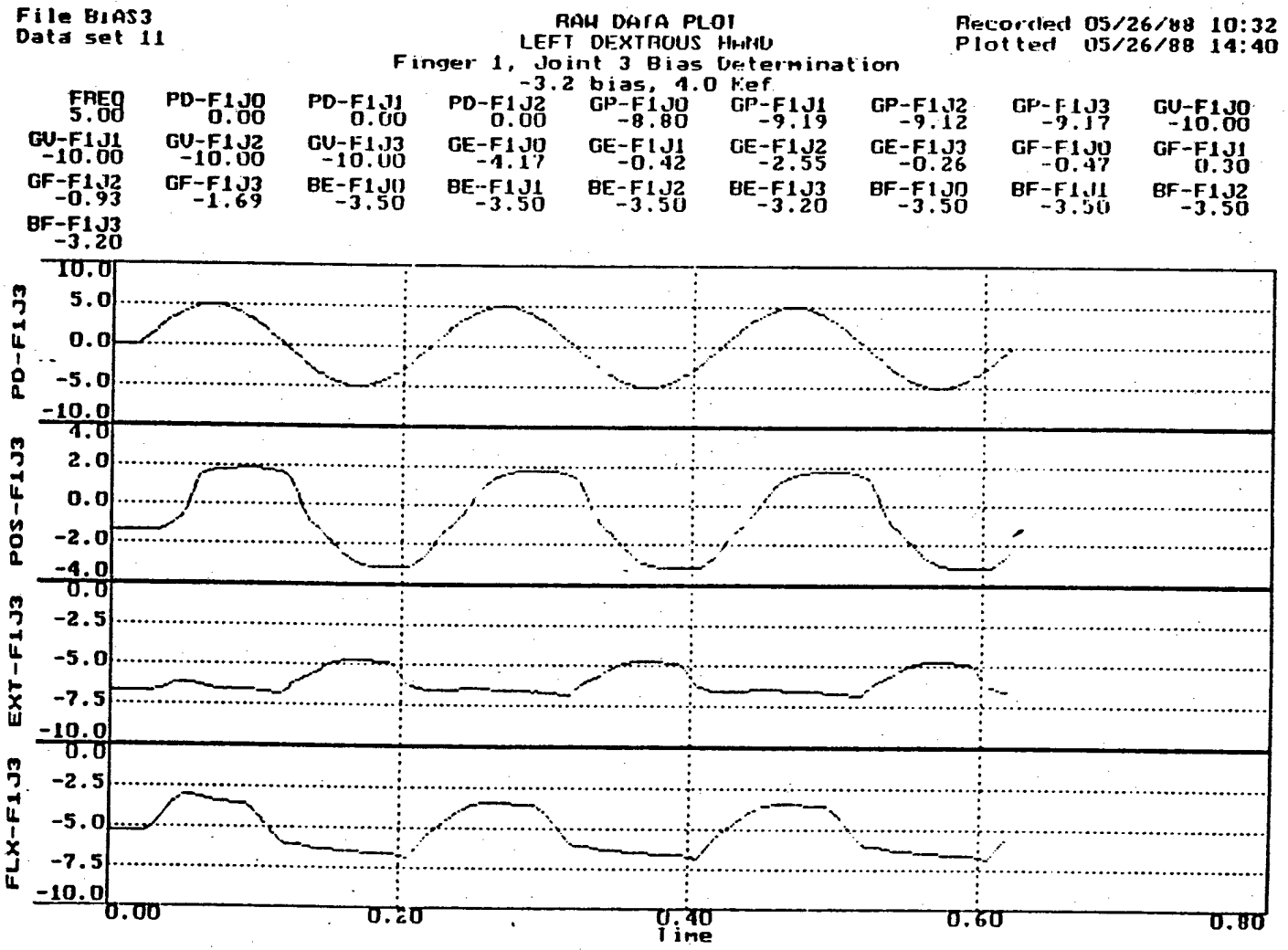



**Figure 4.14 Response of flexion and extension tendons with cocontraction input**

flexion and extension bias setting was thus determined for F1J3. This procedure was repeated for F1J0, F1J1 and F1J2. The bias was determined to be between –4 volts and –3.5 volts. To maintain uniformity and to ensure that snapping never takes place, the flexion and extension bias was set to the maximum of these two values i.e. at –3.5 volts for all the joints.

Once the calibration of various gain blocks of the analog controller and those of the position and tendon tension sensors was carried out successfully, the modeling of the various joints of finger 1 and coupling between them was performed as is discussed in the next chapter.

# V . TRANSFER FUNCTION IDENTIFICATION

This chapter describes the process to identify the transfer functions of the four joints of finger 1 as well as those of the coupling between joints 1, 2 and 3.

## 5.1 INTRODUCTION

The servo loop for each joint of the DH consists of three loops, two inner loops and one outer loop, as shown in the block diagram of figure 5.1. The two inner loops maintain the flexion and extension tendon tensions and the outer loop controls the position of the joint. To identify the open-loop transfer function of each joint, both the force loops were closed and the outer position loop was opened. To check the linearity of the output with the changes in the input amplitude, sine waves of frequency 0.5 Hz. and of varying amplitudes were applied to the servo loop of F1J1 (finger 1 joint 1) and the outputs (POS in figure 5.1) were recorded. This test was repeated using sine waves of frequency 2 Hz. The results are shown in figures 5.2 and 5.3 and the tabulated results are given in Appendix C. The various gain block and bias settings are shown on top of the plots in figures 5.2 and 5.3. It can be seen from these figures and Table C1 in Appendix C that

1. the peak-to-peak voltage of the output increased in a nonlinear fashion with the increase in the input peak-to-peak voltage.

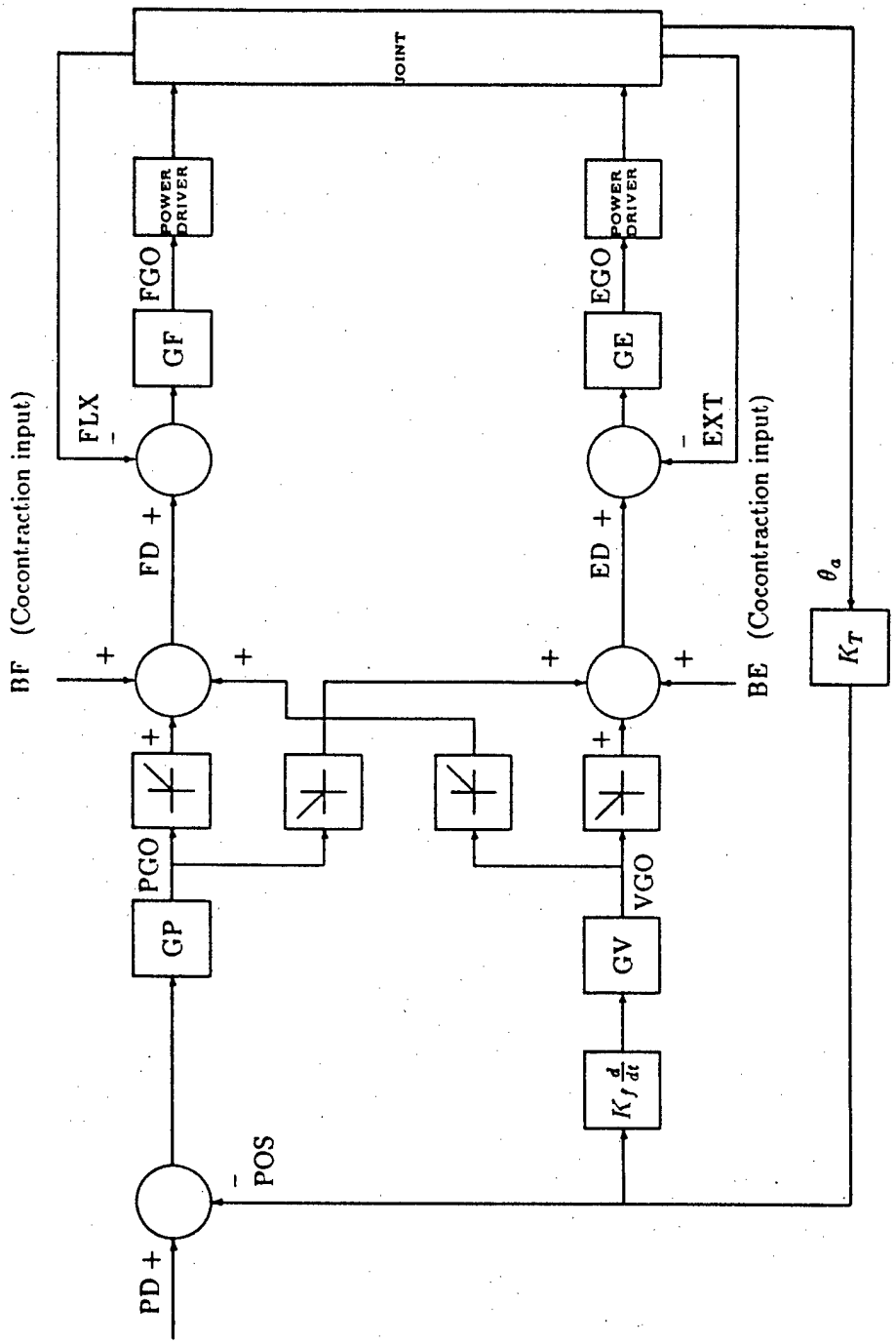



**Figure 5.1 Block diagram of the position control loop of a single joint showing the various blocks to be calibrated**

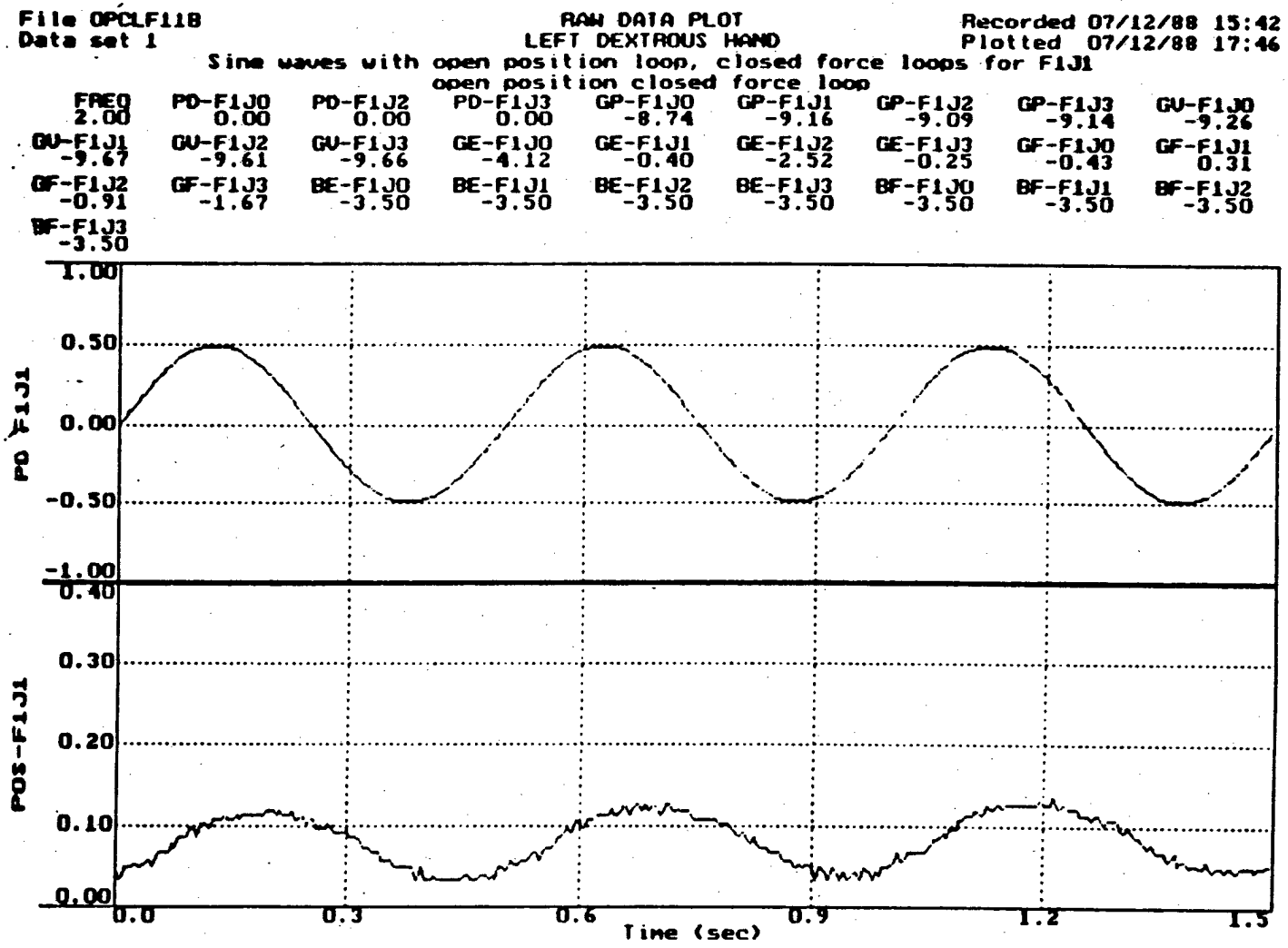



**Figure 5.2 Response of finger 1 joint 1 for a sine wave of amplitude 1 volt**

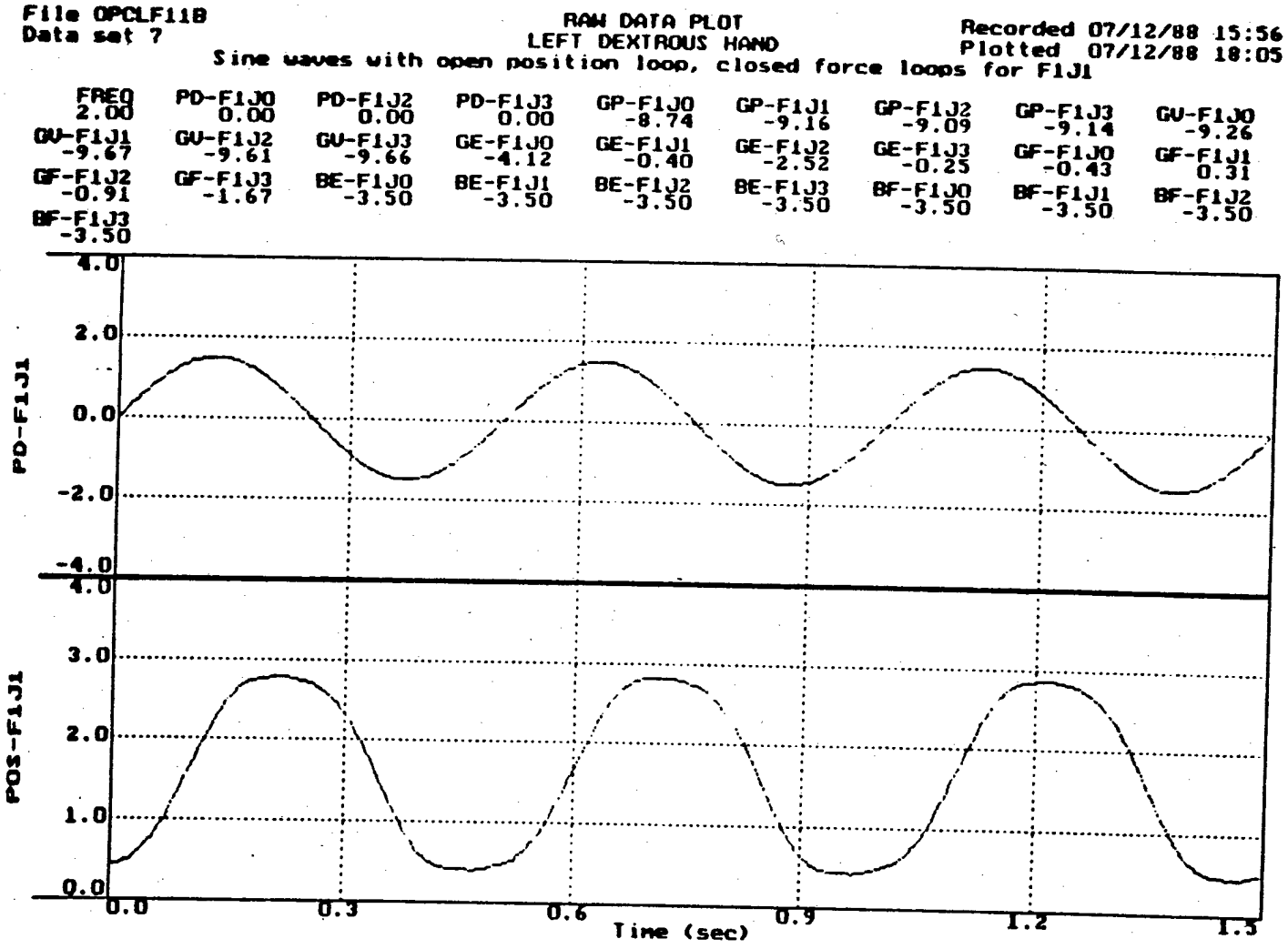



**Figure 5.3** Response of finger 1 joint 1 for a sine wave of amplitude 2 volts

2. as the input voltage increased, the output was flexing more than extending (figures 5.2 and 5.3). This was found to be due to the absence of position control loop.

One of the reasons for the nonlinearities may be the differences in the dynamics of the flexion actuation package and the extension actuation package. Due to the above-mentioned reasons, it was decided to close the position loop when conducting the frequency response tests for identifying the open-loop transfer functions of the various joints.

## 5.2 NOTATION

Figure 5.1, as mentioned above, shows the block diagram of the servo loop of a single joint. Notations used in the figure and their explanations are as follows :

| | | |
|---|---|---|
| PD | – | Position desired in volts (input to the servo loop) |
| POS | – | Position output of the Hall effect sensor in volts (output of the servo loop) |
| PGO | – | Output of the position gain block in volts |
| EXT | – | Output of the extensor tendon tension sensor in volts |
| FLX | – | Output of the flexor tendon tension sensor in volts |
| GP | – | Gain setting of the position gain block in volts |
| GV | – | Gain setting of the velocity gain block in volts |
| GE | – | Gain setting of the extension gain block in volts |
| GF | – | Gain setting of the flexion gain block in volts |

$G_{xx}(s)$ – Open loop transfer function of finger x

$T_{xx}(s)$ – Closed loop transfer function of finger x joint x

$TO_{xy}(s)$ – transfer function of coupling between the outputs of joint x and joint y due to input at joint x

$T_{xy}(s)$ – transfer function of coupling between input of joint x and output of joint y due to input at joint x

$X_x(s)$ – input of joint x

$Y_x(s)$ – output of joint x

$K_p$ – proportional constant of the PID controller

$K_v$ – differential constant of the PID controller

$K_f$ – Gain factor of the differentiator circuit

$K_i$ – integral constant of the PID controller

## 5.3 IDENTIFICATION OF TRANSFER FUNCTION OF FINGER 1 JOINT 0

The objective of this experiment is to determine the transfer function of joint 0 of finger 1. Sinusoidal inputs of different frequencies were applied to the position servo loop of finger 1 joint 0 and the output was measured. From this information, gain and phase plots were obtained. Then, by fitting straight lines to the gain and phase plots, the transfer function of the system was determined. The procedure of identification is explained in more detail in the following subsections. The hand was placed on the back of its palm as shown in figure 3.3 so that the plane of motion of F1J0 (finger 1 joint 0) was parallel to the ground. This was done to reduce the effect of gravity since

the gravitational force will be constant throughout the time of operation of F1J0.

### 5.3.1 INITIAL SETTINGS

1. The gains of the position gain block GP-F1J0, GP-F1J1, GP-F1J2, GP-F1J3 were set to one.

2. The gains of the velocity gain block GV-F1J0, GV-F1J1, GV-F1J2, GV-F1J3 were set to zero. This was done because the objective of this experiment is to find the model of the joint without the inclusion of any control parameters.

3. The gains of the flexion gain block GF-F1J0, GF-F1J1, GF-F1J2 and GF-F1J3 were set to four. This was done to ensure that the force loop has considerable stiffness, and that, it will regulate the tendon tensions.

4. The gains of the extension block GE-F1J0, GE-F1J1, GE-F1J2 and GE-F1J3 were set to four for the same reasons as mentioned for the flexion gain block.

5. The flexion tendon biases BF-F1J0, BF-F1J1, BF-F1J2 and BF-F1J3 and extension tendon biases BE-F1J0, BE-F1J1, BE-F1J2 and BE-F1J3 were set to −3.5 volts as determined in Chapter 4.

The parameters of F1J1, F1J2 and F1J3 can be set to any value since they do not affect the response of F1J0. Two joints are said to be coupled if the movement of one joint results in the movement of the other joint. There is no coupling between joint 0 and joints 1, 2 and 3 since the axis of motion of joint 0 is perpendicular to those of joints 1, 2 and 3 (figure 2.6).

### 5.3.2 PROCEDURE

The position loop and flexion and extension force loops were closed. Joints 1, 2 and 3 were commanded to full extension during the entire experiment. A sinusoidal voltage signal (PD in figure 5.1) was applied at the input of the servo loop of finger 1 joint 0 and the data was recorded at the following frequencies :

0.1 – 3.0 Hz. increasing in steps of 0.2 Hz.

3.0 – 7.0 Hz. increasing in steps of 0.5 Hz.

7.0 – 30.0 Hz. increasing in steps of 2 Hz.

The following parameters namely position output (POS-F1J0), output of the position gain block (PGO-F1J0), output of the flexion gain block (FGO-F1J0), output of the extension gain block (EGO-F1JO), flexion tendon tension (FLX-F1JO) and extension tendon tension (EXT-F1J0) (figure 5.1) were recorded for each of the above frequencies. The frequency was incremented in smaller steps in the low frequency region because the frequency range of interest for the identification of the transfer function of the joint is 0.1 – 7.0 Hz. The open loop transfer function of the joint can be obtained by considering PGO-F1J0 (figure 5.1) as the input and the position of the joint (POS-F1J0) as the output. The gain of the system was calculated by dividing the magnitude of the output (POS-F1J0) by the magnitude of the error between position desired (PD-F1J0) and actual position (POS-F1J0), which is same as output of the position gain block (PGO-F1J0 in figure 5.1) since the gain GP-F1J0 was set to one. The gain in dB and the phase angle in degrees between the input and the output was calculated. The details of the calculations are given in the next

section.

### 5.3.3 PROGRAM TO CALCULATE GAIN AND PHASE

The program to calculate the gain in dB and the phase in degrees is given in Appendix D. The program was written in $MATRIX_x$, which is a powerful computer-aided design package. The amplitudes of the input sine wave (PGO-F1J0) and the output sine wave (POS-F1J0) were computed for all the above mentioned frequencies. The gain in dB was calculated using the formula

$$\text{gain} = 20 \log \left( \frac{\text{output}}{\text{input}} \right) \text{db} \tag{5.1}$$

The phase of the output was calculated using the formula:

$$\text{phase} = \frac{t_{out} - t_{in}}{t_p} \times 360 \text{ degrees} \tag{5.2}$$

where

$t_{in}$ – time at which the input sine wave (PGO-F1J0) crosses the reference line (0 volts)

$t_{out}$ – time at which the output sine wave (POS-F1J0) crosses its reference line

$t_p$ – time period of the sine wave which is obtained by calculating 1/frequency (Hz.)

The gain (in dB) and phase (in degrees) was tabulated and plotted against the frequency of oscillation $\omega$ in radians/sec. Using straight line approximations (Bode plot) on the gain and phase plots, the transfer function of the joint was determined.

The transfer function is of second order. Since there was a steady-state error for a step input, the type of the system was identified as 0. The general form of the transfer function is given by

$$G(s) = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{5.3}$$

Closing the position servo loop with a unity feedback, the closed-loop transfer function is obtained which is given by

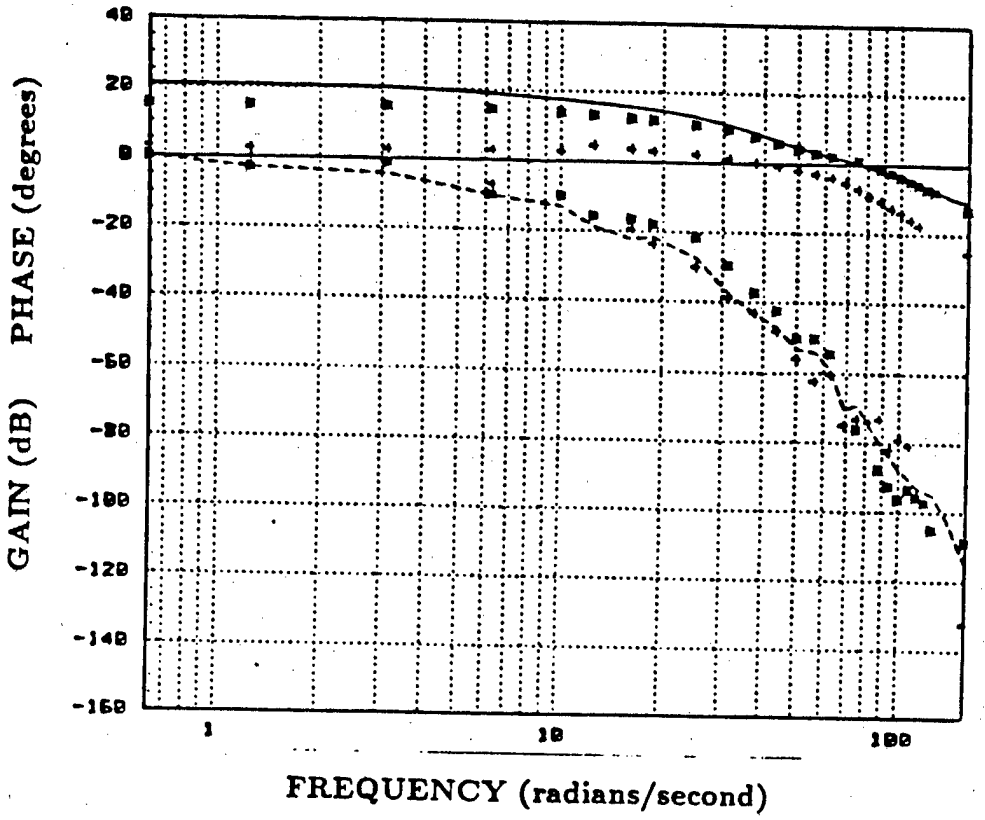
$$\frac{C(s)}{R(s)} = \frac{K\omega_{ncl}^2}{s^2 + 2\zeta_{cl}\omega_{ncl} s + \omega_{ncl}^2} \tag{5.4}$$

where

$$\omega_{ncl} = \omega_n(\sqrt{1+K}), \qquad \zeta_{cl} = \frac{\zeta}{\sqrt{1+K}} \tag{5.5}$$

The gain and phase was calculated for sinusoidal inputs of 2V, 5V and 8V. Figure 5.4 shows the gain-phase plot of F1J0 for sinusoidal inputs of amplitudes 2V, 5V and 8V. In all the three cases, the roll-off started pretty much at the same frequency. The only difference was in the gain of the system K. K increased with increase in amplitude (figure 5.4). It can be seen from equation ( 5.5) that $\zeta_{cl}$ is inversely proportional to K. Thus, with increase in amplitude, $\zeta_{cl}$ decreases which makes the system more underdamped. Hence, it was decided to design the controller for the 8V sine wave. This might result in making the system which is critically damped for 8V, slightly overdamped for inputs of lesser amplitude. But an overdamped system is preferred to an underdamped system.

Figure 5.5 shows the gain-phase plot of F1J0 for sinusoidal inputs of amplitude 8V. It can be seen from the figure that the phase plot starts at -56.16 degrees. Initially, it was thought to be a constant phase shift which translates to a variable delay in the



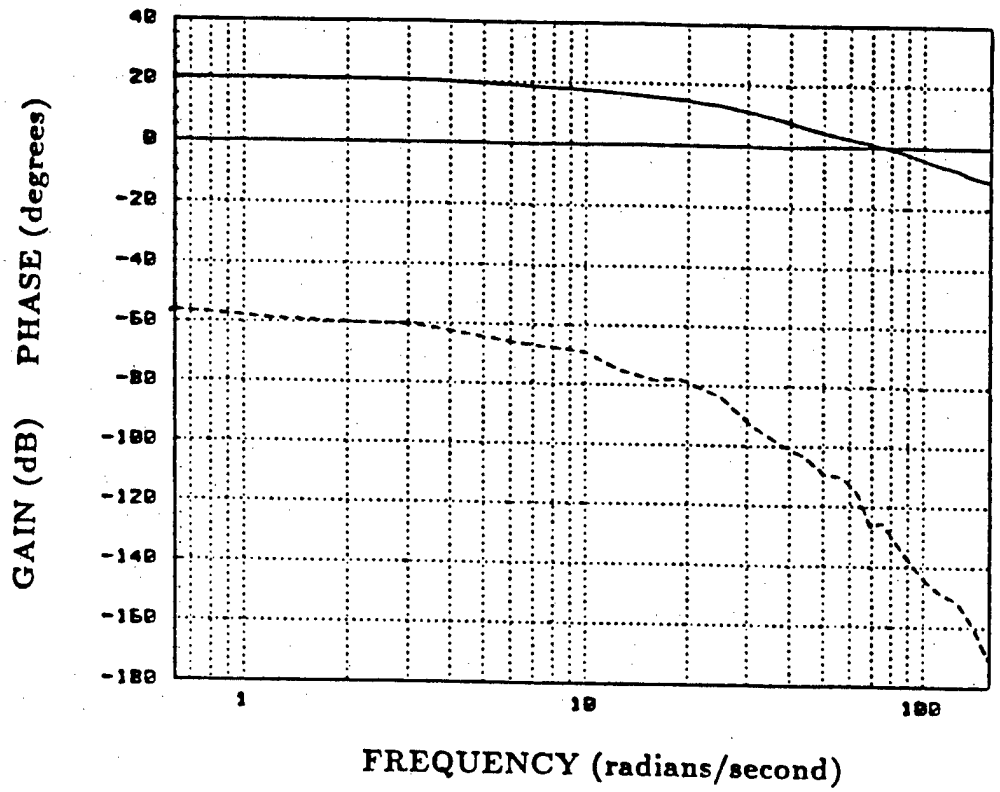

——— gain plot for sinusoidal inputs of amplitude 8 volts

- - - phase plot for sinusoidal inputs of amplitude 8 volts

* * * gain and phase plots for sinusoidal inputs of 5 volts

+++ gain and phase plots for sinusoidal inputs of 2 volts

**Figure 5.4 Gain and phase plots of finger 1 joint 0**




gain plot of finger 1 joint 0

phase plot of finger 1 joint 0

**Figure 5.5 Gain and phase plot of finger 1 joint 0**

time domain. However, after additional experimentation, it was found that when the position gain $K_p$ was increased, the initial phase shift disappeared. This can be seen from figures 5.6 and 5.7. Thus, the initial phase shift can be ignored, since $K_p$ was designed to have a high value ( design discussed in section 5.4). The transfer function of joint 0 was determined to be given below:

$$G_{00}(s) = \frac{9533.25}{s^2 + 240\,s + 3600} \tag{5.6}$$

Figure 5.8 shows the gain-phase plot of F1J0 as well as that of the estimated model.

The procedure for identification of the plant using sinusoidal inputs of 2V, 5V and 8V was repeated for joints 1, 2 and 3 of finger 1. The only difference is that there is coupling between joints 1, 2 and 3 . The coupling exists between a particular joint and joints distal to it only i.e. joint 2 moves when joint 1 is commanded to move, due to coupling but joint 1 does not move when joint 2 is commanded to move. Hence, when the joint under consideration was commanded to move, the tendons of all distal joints were disconnected. Joints 1, 2 and 3 were initially determined to be of second order, with a general transfer function as follows :

$$G(s) = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} = \frac{K\omega_n^2}{(s + p_1)(s + p_2)} \tag{5.7}$$

where $p_1, p_2$ are the poles of G(s).

However, the step responses of the joints 1, 2 and 3 were highly oscillatory and were tending to remain so when the position error gain GP was increased (Appendix C). This could not happen if the joints were of second order. Hence, the transfer functions of joints 1, 2 and 3 were determined to be of third order. The transfer

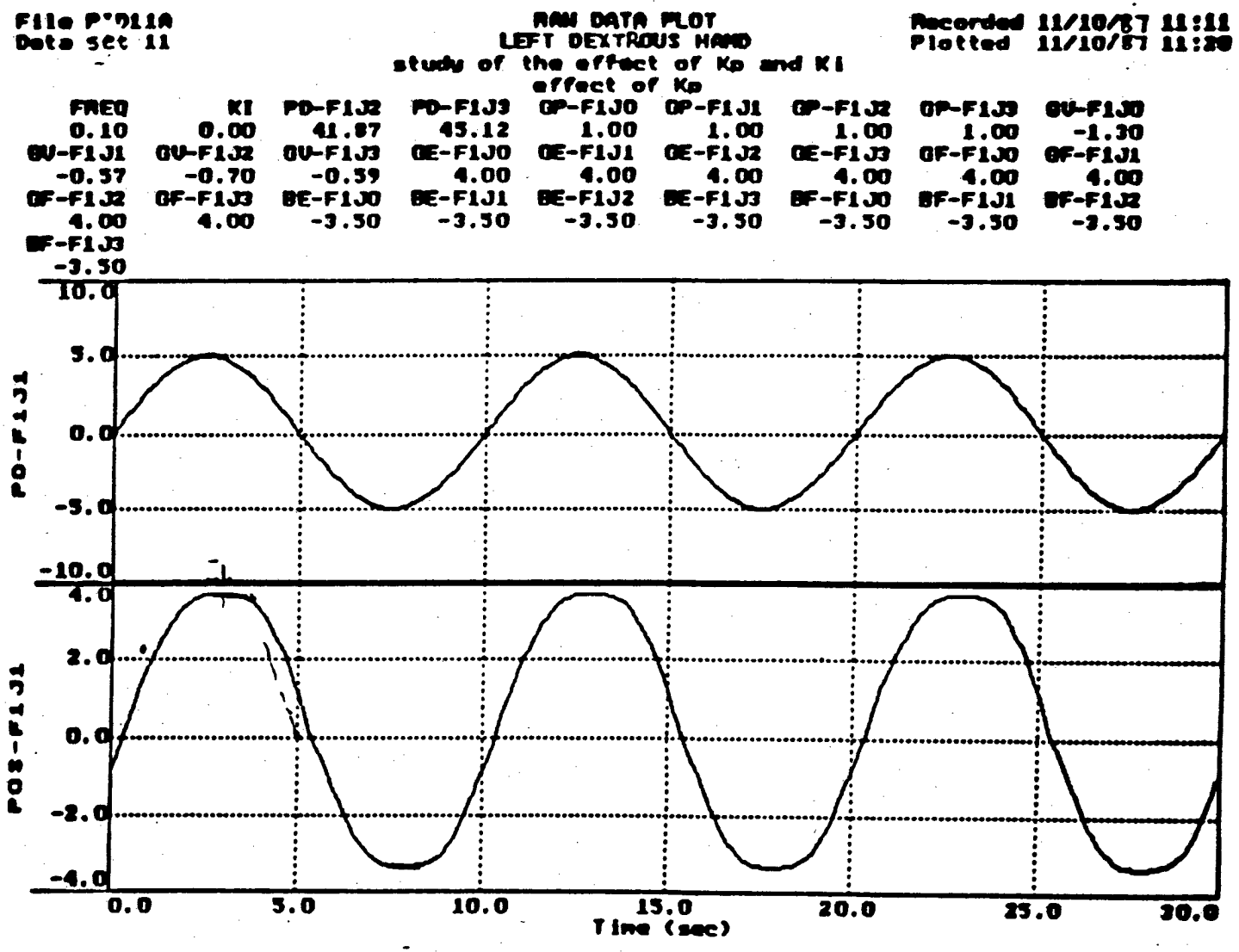



Figure 5.6 Response of finger 1 joint 1 for $K_p = 1$

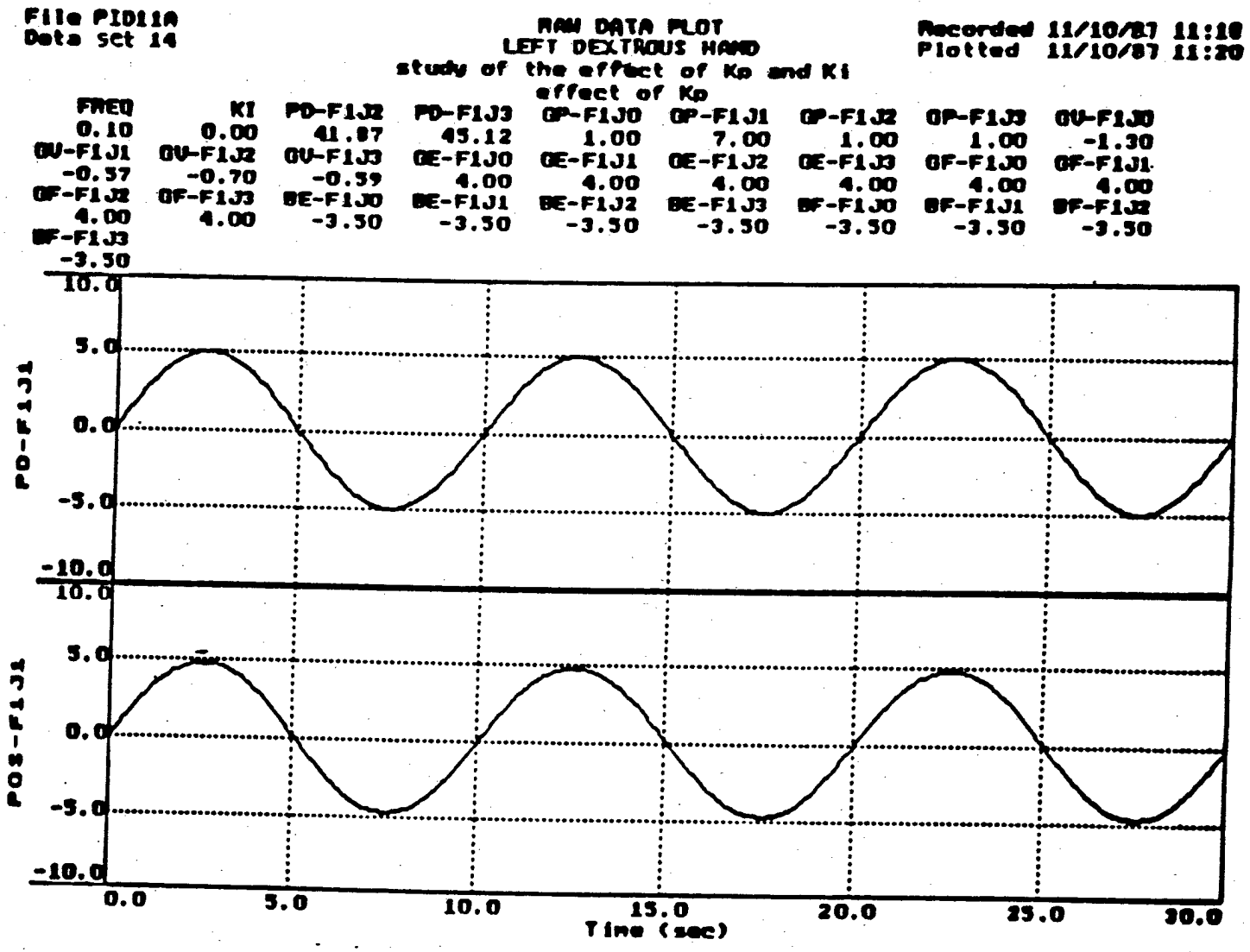
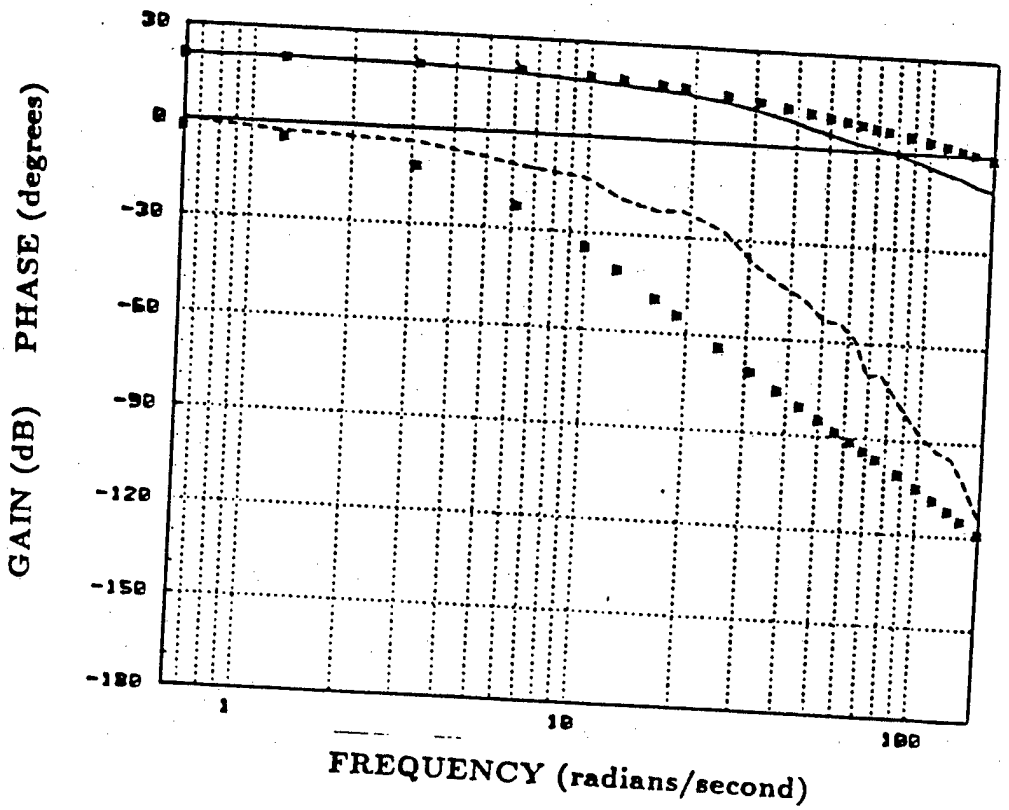



Figure 5.7 Response of finger 1 joint 1 for $K_p = 7$




——— gain plot obtained from experimental data

- - - phase plot obtained from experimental data

* * * gain and phase plots of the estimated model

**Figure 5.8 Gain and phase plots of finger 1 joint 0 and its estimated model**

functions of joints 1,2 and 3 are as follows :

Finger 1 Joint 1

$$G_{11}(s) = \frac{6468750}{(s^2 + 245s + 3450)(s + 600)} \tag{5.8}$$

Finger 1 Joint 2

$$G_{22}(s) = \frac{2986665}{(s^2 + 90s + 5625)(s + 300)} \tag{5.9}$$

Finger 1 Joint 3

$$G_{33}(s) = \frac{14230352}{(s^2 + 340s + 22500)(s + 800)} \tag{5.10}$$

The gain-phase plots and tables of joints 1, 2 and 3 of finger 1 and of their estimated models are given in Appendix C.

## 5.4 DESIGN OF INDIVIDUAL JOINT LOOPS

From equation( 5.6), it can be seen that joint 0 is a second order system. Joints 1, 2 and 3 (equations( 5.8), ( 5.9) and ( 5.10) ) are determined to be third order systems with two quadratic poles and a simple pole. Since the simple pole is far away in the left half plane, it can be ignored while carrying out the design steps. Hence, all the four joints can be modelled as second order systems with transfer functions of the form

$$G_{pp}(s) = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{5.11}$$

where $\omega_n$ is the natural frequency and $\zeta$ is the damping ratio of the system.

Figure 5.9 shows the block diagram of the PD (proportional plus derivative) control loop of an individual joint. The closed loop transfer function of figure 5.9 can be written as

$$\frac{Y_p(s)}{X_p(s)} = \frac{KK_p\omega_n^2}{s^2 + 2\zeta_{cl}\omega_{ncl}s + \omega_{ncl}^2} \tag{5.12}$$

where

$$\zeta_{cl} = 2\zeta\omega_n + KK_vK_f \tag{5.13}$$

$$\omega_{ncl} = \sqrt{\omega_n^2\,(1 + KK_p)} \tag{5.14}$$

Since the open loop transfer functions of the joints equation( 5.11) are of type zero, there will be a steady-state error to a step input (figure 5.10). The steady-state error is given by

$$e_{ss} = \frac{R}{1 + KK_p} \tag{5.15}$$

where R is the amplitude of the step input

Since the steady-state error is inversely proportional to $K_p$, the error can be reduced by choosing a high value of $K_p$. However, the highest value of $K_p$ that can be chosen using the analog controller is 10, so $K_p$ was set to 8. An overshoot is undesirable, hence the closed loop transfer function is to be designed to have a damping ratio ($\zeta_{cl}$) of 1. Substituting $K_p = 8$ and $\zeta_{cl} = 1$ in equations ( 5.12), ( 5.13) and ( 5.14), the value of $K_v$ was calculated. The design parameters for the various joints of finger 1 are given in table 5.1

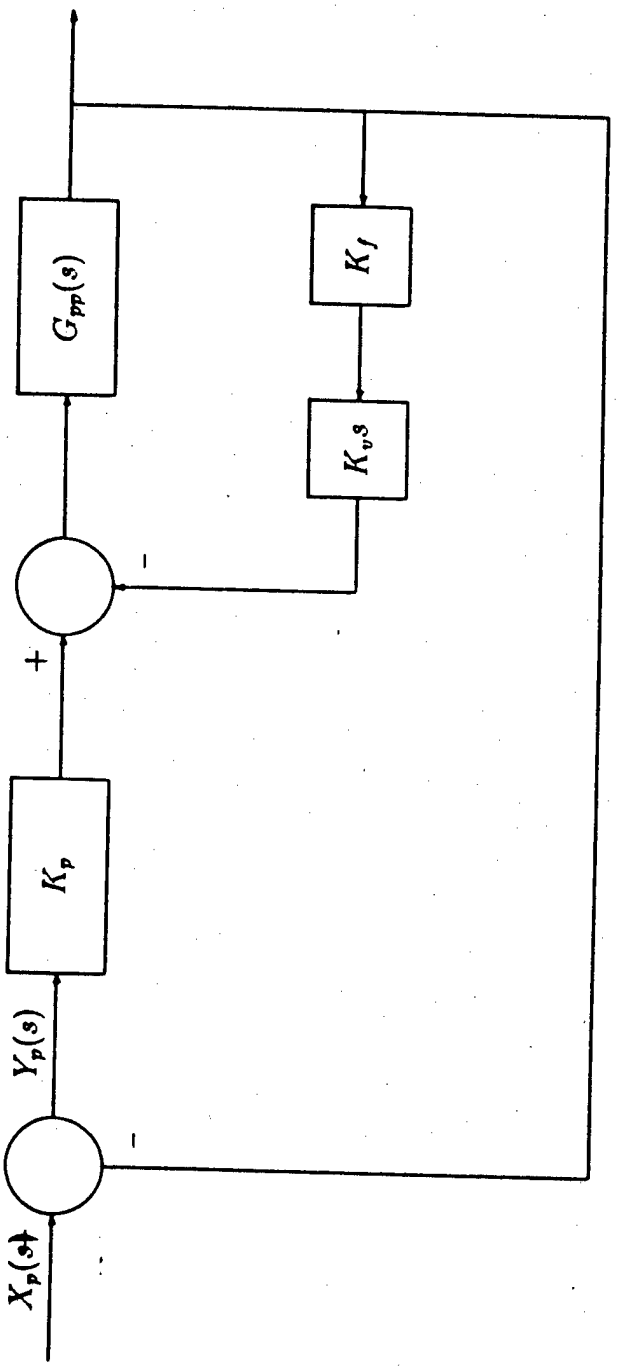


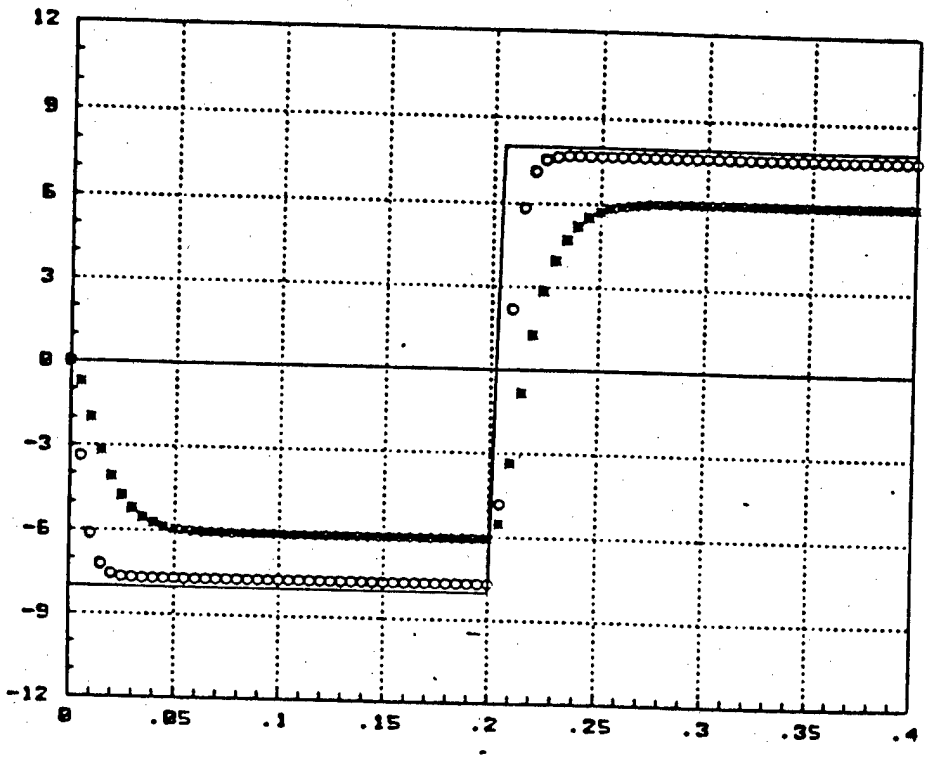

**Figure 5.9 Block diagram of closed-loop position control of a single joint**

**Table 5.1 Design Parameters**

| Joint | $K_p$ | $K_v$ |
|---|---|---|
| Joint 0 | 8 | 4.2 |
| Joint 1 | 8 | 4.1 |
| Joint 2 | 8 | 6.1 |
| Joint 3 | 8 | 3.2 |

## 5.5 INTEGRAL CONTROL

Figure 5.10 shows the step responses of finger 1 joint 0 for designed values of $K_p$ and $K_v$ and for $K_p = 1$ and $K_v = 0$. It can be seen that the step response for the designed values of $K_p$ and $K_v$ has an improved rise time and also the steady-state error is considerably less as compared to the step response for $K_p = 1$. However, there is still a small steady-state error. This is because joint 0 is a type zero system. If the type of the system is increased by one, the steady-state error will be reduced to zero for unit step input[6]. This can be achieved by adding an integral term to the PD controller, thus making it a PID (proportional plus integral plus derivative) controller (figure 5.11). The effect of the integral term can be seen from figure 5.12 where the steady-state error is reduced to zero for $K_i = 2$ . The effect can be seen more clearly from figures C8 and C9 in Appendix C. The integral term was implemented using software. A program was written in C for the integral term to be introduced in the analog controller. The program is given in Appendix E. Another way of reducing the steady-state error is by introducing a gain external to the position servo loop of an




——— step input

* * * step response for $K_p = 1$, $K_v = 0$

oooo step response for $K_p = 8$, $K_v = 4.1$

**Figure 5.10 Step responses of finger 1 joint 1**

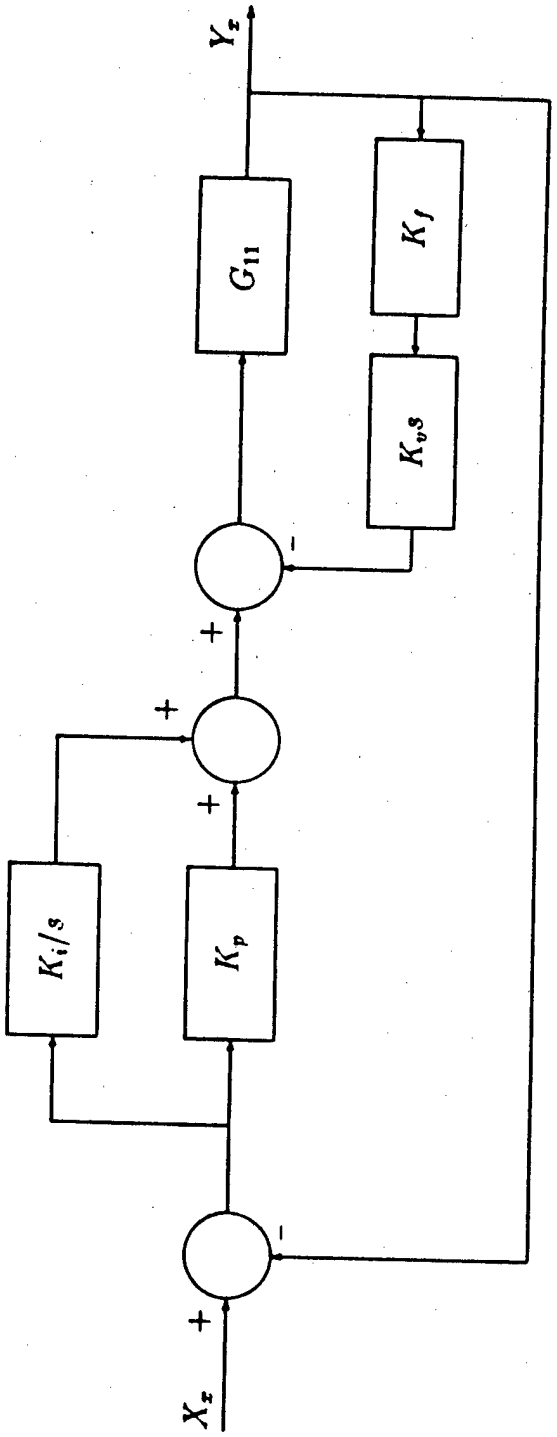
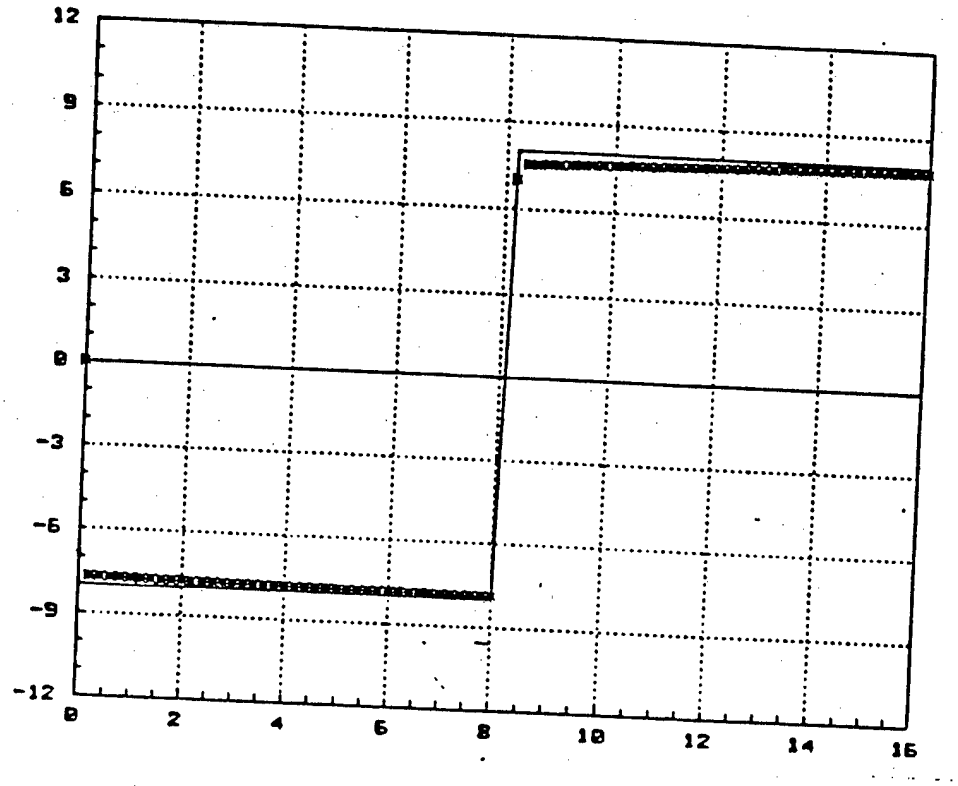



**Figure 5.11 Block diagram of closed loop position control of a single joint**




——— step input

* * * step response with $K_p = 8$, $K_v = 4.1$ and $K_i = 2$

**Figure 5.12 Step response of finger 1 joint 1 with the addition of $K_i$**

individual joint (figure 5.13). The gain $K_g$ is obtained using the formula given below :

$$K_g = \frac{R}{R - e_{ss}} \tag{5.16}$$

The gain $K_g$ for finger 1 joint 1 was determined to be 1.1. The step response of finger 1 joint 1 after the introduction of $K_g$ is shown in figure 5.14. It can be seen that there is no steady-state error.

## 5.6 IDENTIFICATION OF JOINT COUPLING

Two joints are coupled if the motion of one joint affects the motion of the other, i.e. when joint 1 is moved, if joint 2 moves then joints 1 and 2 are coupled.

Given below is a description of the nature of coupling between joints.

1. There is no coupling between joint 0 and the other three joints in the finger. This is because the axis of motion of joint 0 is perpendicular to those of joints 1, 2 and 3.

2. Among the coupled joints (joints 1, 2 and 3), there is a significant coupling between the joint in consideration and the joints distal to it only .

The coupling between the various joints is shown in table 5.2.

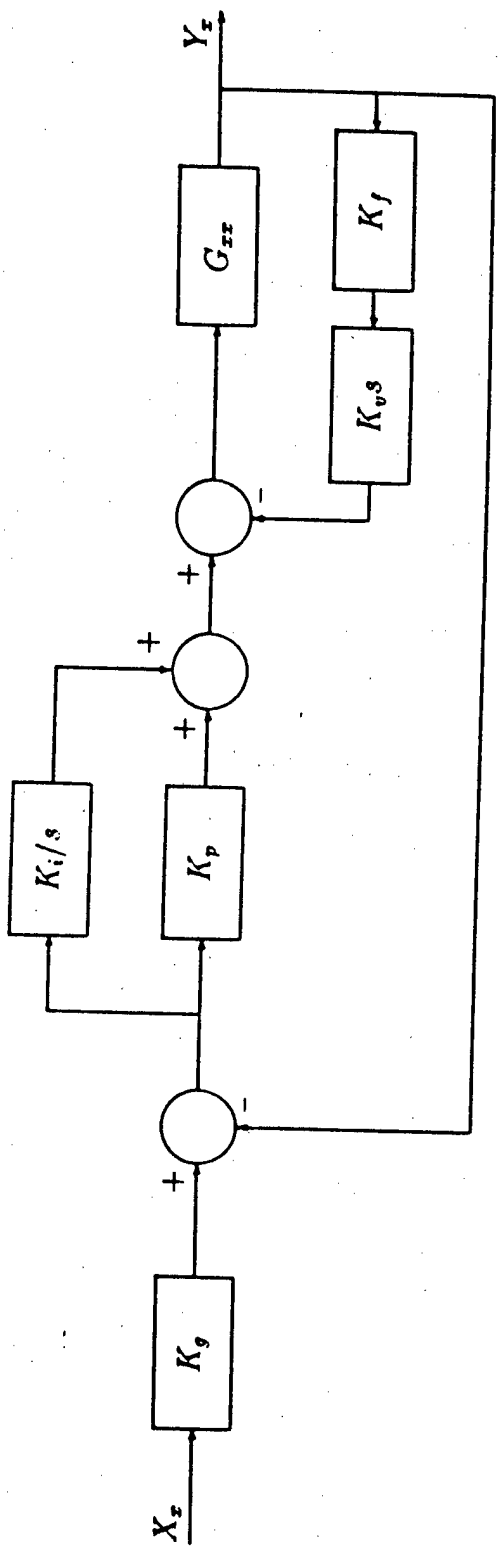
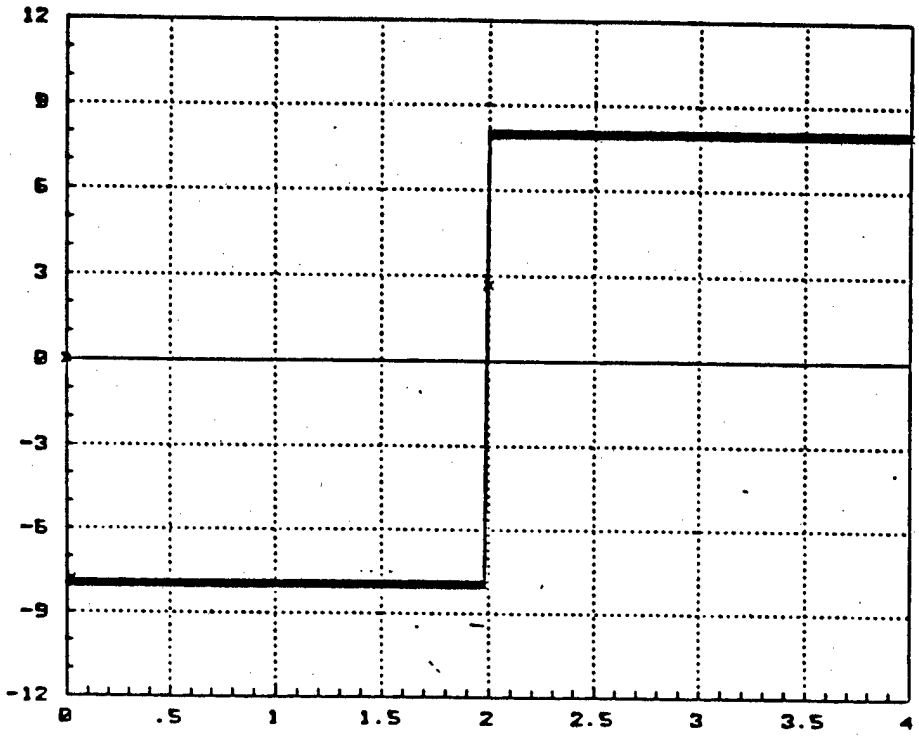



**Figure 5.13 Block diagram of closed loop position control of a single joint**




——— step input

xxxx step response with $K_p = 8$, $K_v = 4.1$ and $K_g = 1.1$

**Figure 5.14 Step response of finger 1 joint 1 with the addition of $K_g$**

Table 5.2 Nature of coupling between joints 1, 2 and 3

| Joint at which the input is applied | Coupled joints | Non-coupled joints |
|---|---|---|
| Joint 0 | None | Joints 1,2&3 |
| Joint 1 | Joint2, Joint3 | Joint 0 |
| Joint 2 | Joint 3 | Joint1,Joint0 |
| Joint 3 | None | Joint0,Joint1,Joint2 |

One of the reasons for the coupling between different joints may be due to the fact that the tendons of different joints are routed over pulleys mounted on the same shaft. For example, the tendons of joints 2 and 3 are routed over pulleys mounted on the same shaft as the pulleys that carry the tendons of joint 1. This may justify the existence of significant coupling between the joint in consideration and distal joints only, thus when joint 1 moves, the flexor and extensor tendons of joints 2 and 3 also move, which results in the movement of joints 2 and 3. A further reason for the existence of the coupling may be due to inertial loading.

Initially, it was not certain whether the coupling is between the input of the joint commanded to move and the output of the coupled joint (input-output coupling) (figure 5.15) or between the outputs of the joint commanded to move and the coupled joint (output-output coupling) (figure 5.16) . After conducting a few test runs, it was concluded that the latter is true. The reasons for reaching this conclusion are as follows :

1. Figures 5.17 and 5.18 show plots of step inputs applied to joint 2 (PD-F1J2)

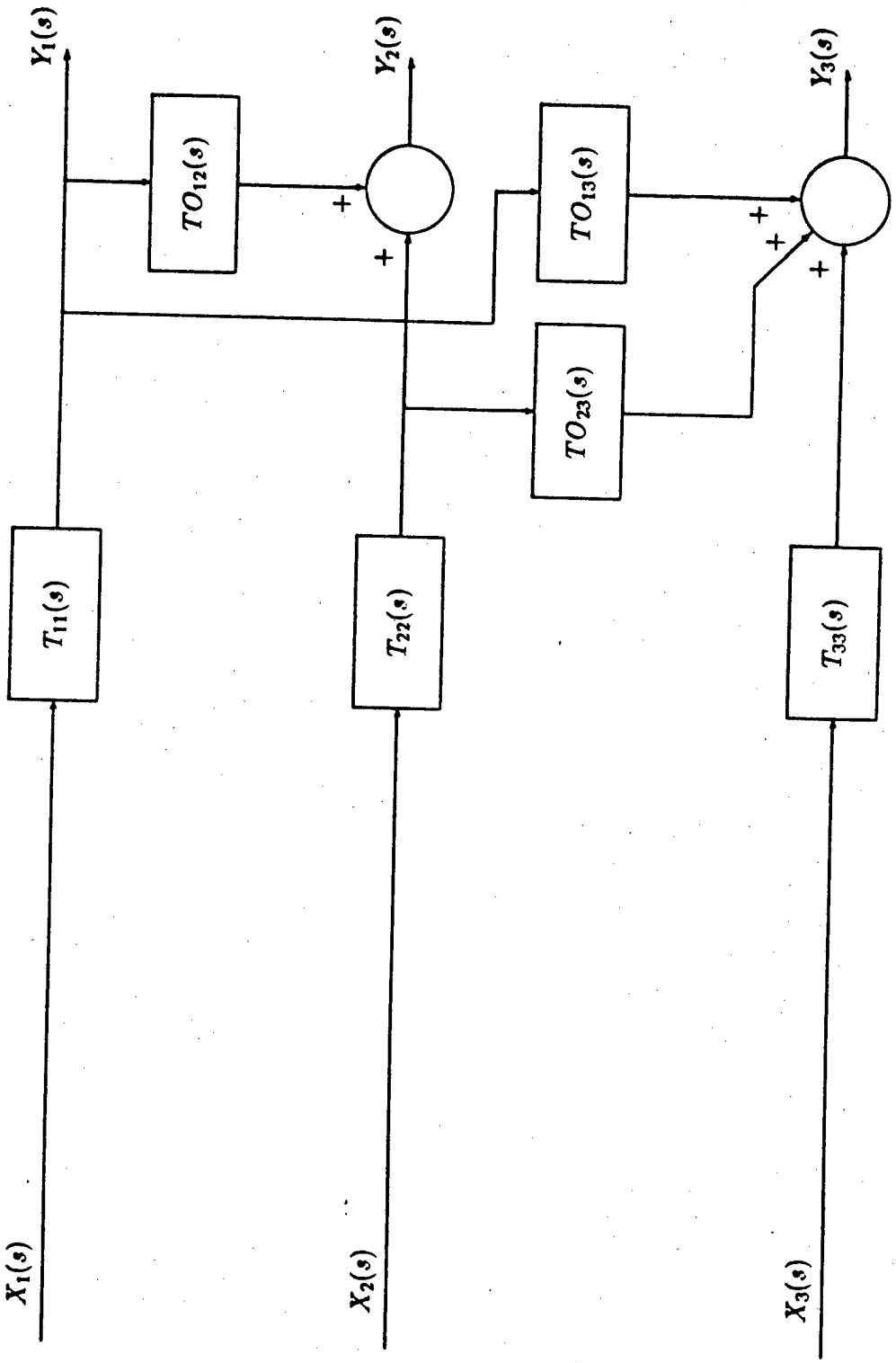



**Figure 5.15 Block diagram illustrating output to output coupling between F1J1, F1J2 and F1J3**

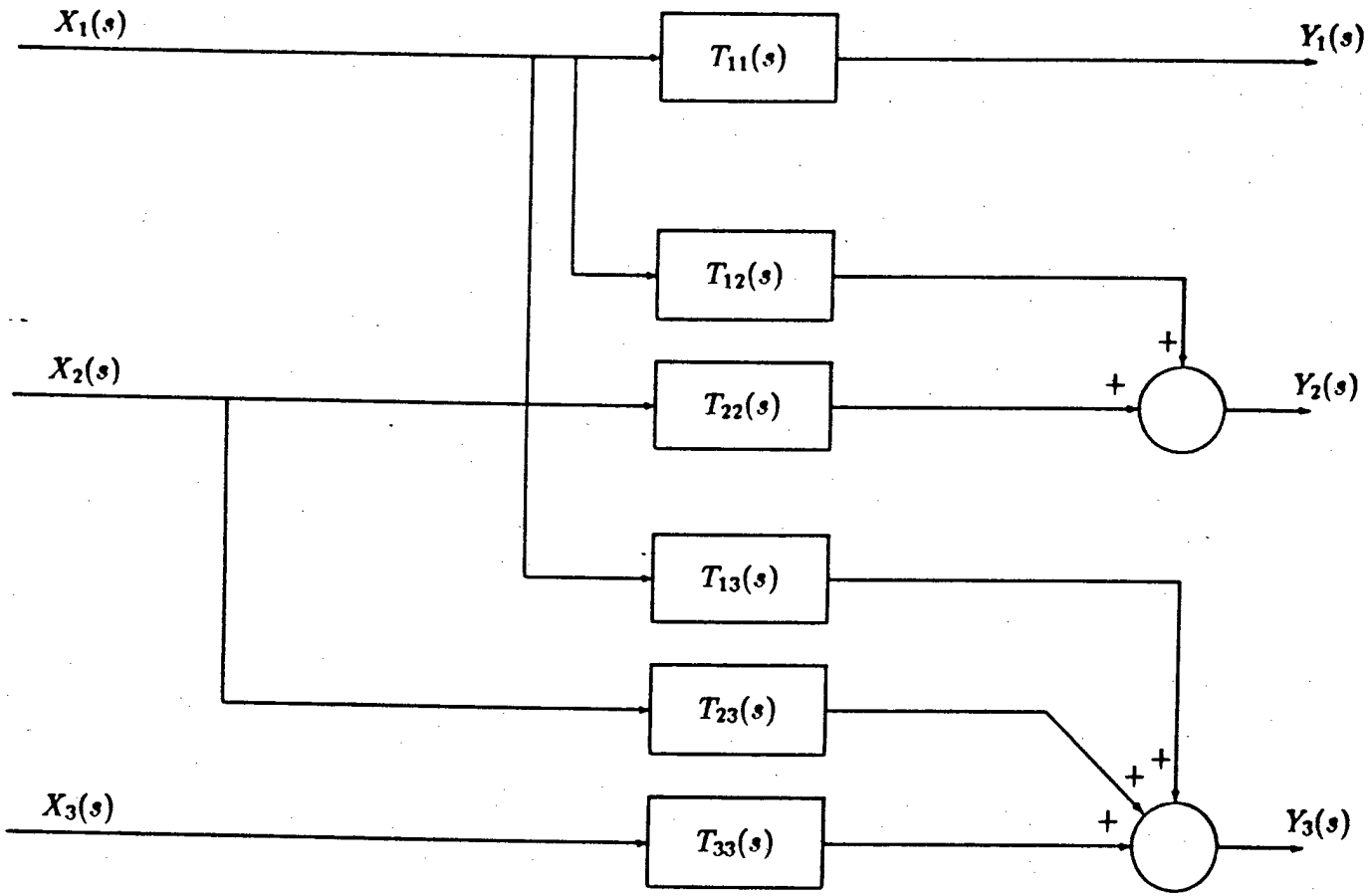



**Figure 5.16 Block diagram illustrating input to output coupling between F1J1, F1J2 and F1J3**

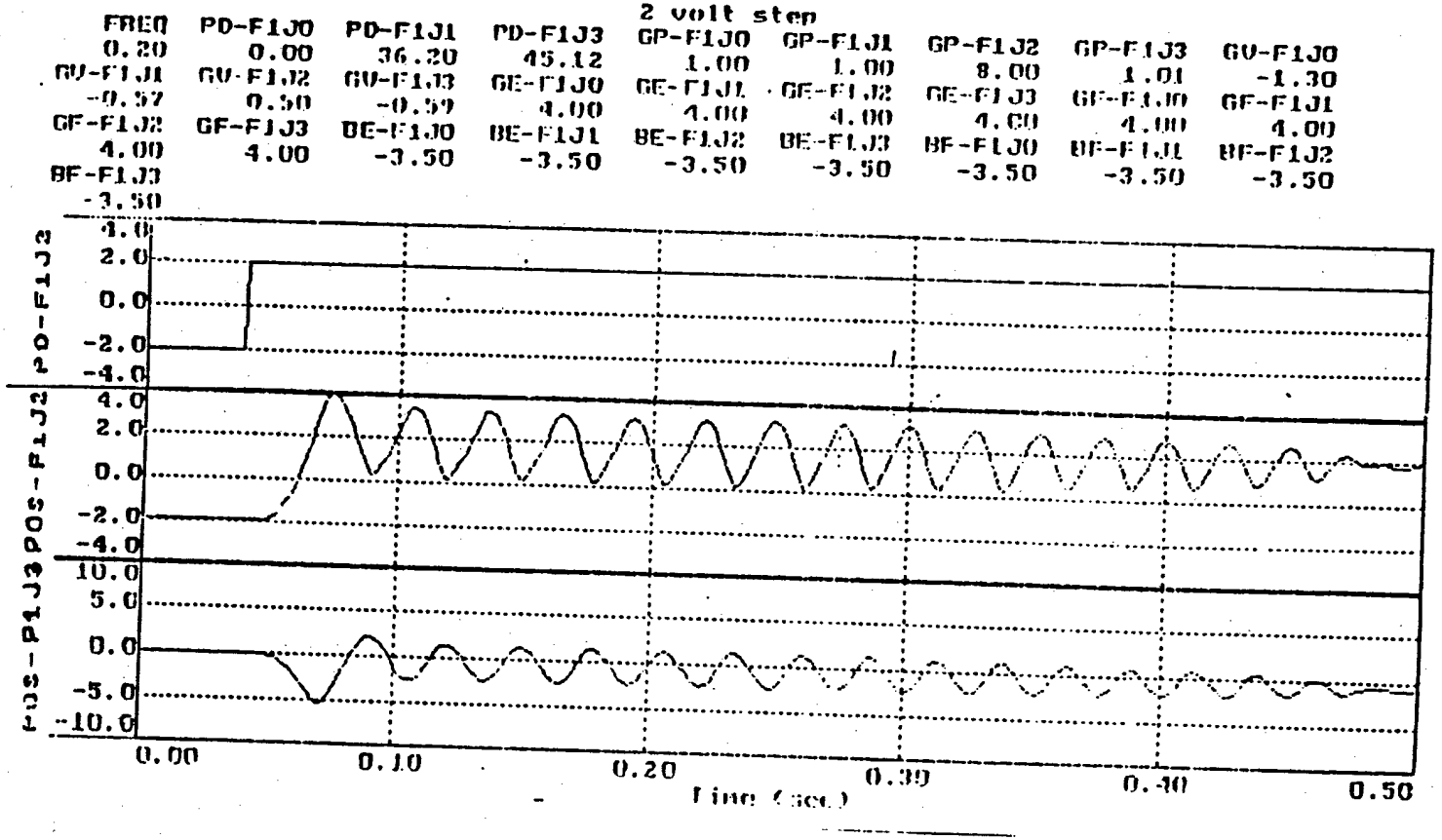



**Figure 5.17 Step response of F1J2 and F1J3 illustrating the nature of coupling between F1J2 and F1J3**

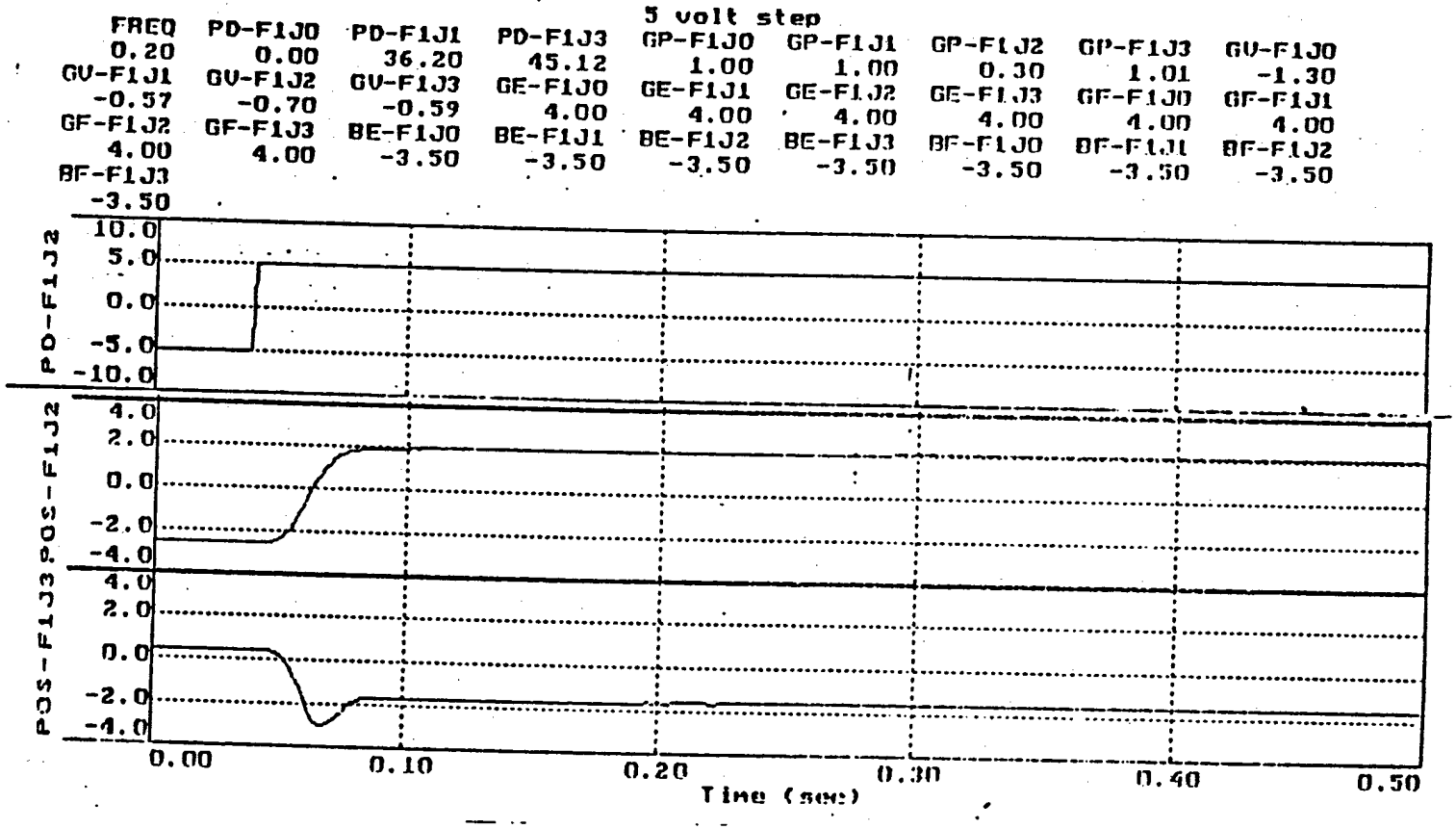



**Figure 5.18 Step response of F1J2 and F1J3 illustrating the nature of coupling between F1J2 and F1J3**

and the step responses of joints 2 and 3 (POS-F1J2 and POS-F1J3) for a gain GP of 0.3 and gain GP of 8 respectively. It can be seen from figure 5.17 that the response of joint 2 settled down to a steady-state value and so did the response of joint 3. However, in figure 5.18, the step response of joint 2 was oscillatory which resulted in joint 3 being oscillatory too. From these two figures, it can be concluded that the output of joint 3 (coupled joint) depends on the output of joint 2 (joint commanded to move).

2. The gain and phase plots of input-output coupling ($T_{xy}(s)$ in figure 5.16) changed with the change in control parameters ($K_p$ and $K_v$ in figure 5.9) of the closed loop control of joint x. This implies that the coupling term $T_{xy}$ depends upon the output of joint x, which confirms the above conclusion.

## 5.7 COUPLING TRANSFER FUNCTIONS

The transfer functions of coupling between joints were obtained using the frequency response analysis. The position and force loops of all the joints were closed. Given below is the procedure for the identification of coupling between joints 1 and 2 due to an input at joint 1. Sinusoidal inputs of amplitude 8V were applied at the input of the servo loop of joint 1 (PD in figure 5.1) for the following frequencies :

0.1–3.0 Hz. increasing in steps of 0.2 Hz.

3.0–7.0 Hz. increasing in steps of 0.5 Hz.

7.0–30.0 Hz. increasing in steps of 2 Hz.

The position outputs of joints 1, 2 and 3 (POS-F1J1, POS-F1J2 and POS-F1J3)

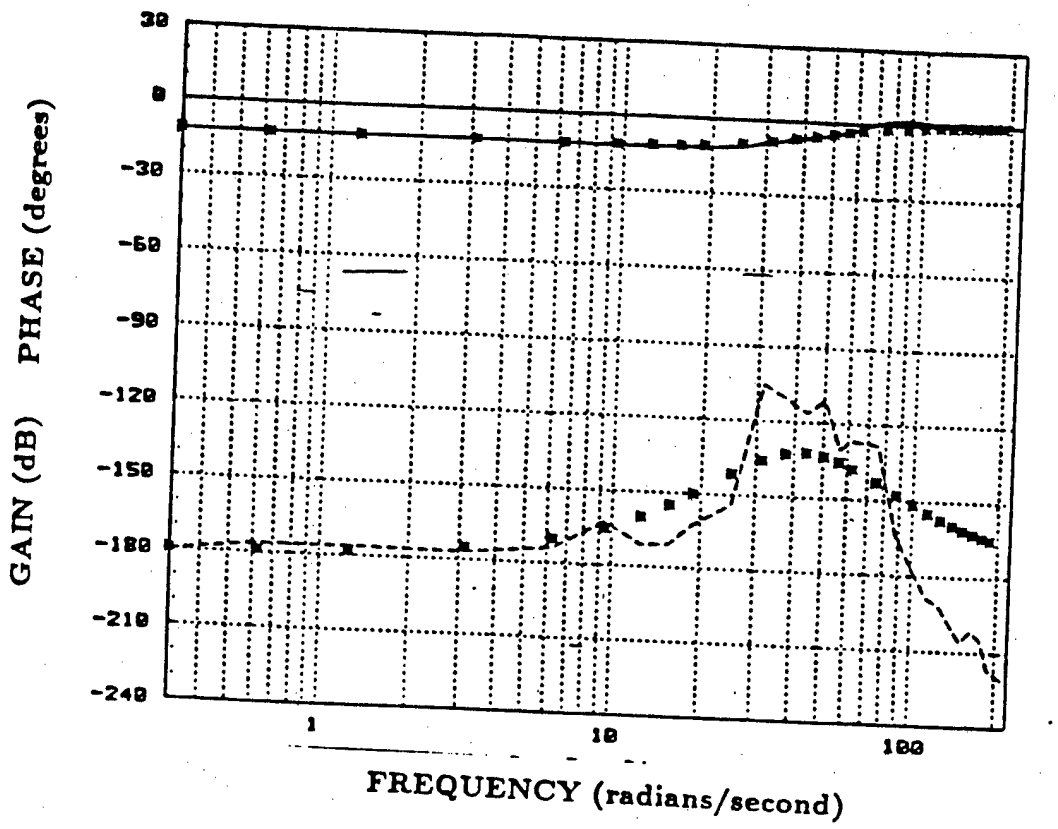
were recorded. A program was written in MATRIX$_x$ to calculate the gain and phase with POS-F1J1 as input and POS-F1J2 as output. This way, the transfer function of coupling between the output of joint 1 and output of joint 2 (output-output coupling) can be determined. This program was written in a similar manner as the program written to identify the transfer functions of individual joints . The frequencies were increased in smaller steps in the lower frequency region, because the frequency range of interest for identification of the transfer functions of coupling between various joints is 0.1 - 7.0 Hz. However, the data was collected for frequencies upto 30 hertz since the bandwidth of each joint was determined to be in the range between 15 and 20 hertz. The program to calculate the gain and phase is given in Appendix D. The gain and phase were plotted against frequency $\omega$ (radians/second). The transfer function of the coupling was obtained by using straight-line approximations on the frequency response plot. The transfer function $TO_{12}$ is given by

$$TO_{12}(s) = -\frac{s^2 + 48s + 900}{s^2 + 96s + 3600} = -\frac{s + 24 \pm j18}{s + 48 \pm j36} \tag{5.17}$$

In the same manner, the coupling between joints 1 and 3 ($TO_{13}$), and joints 2 and 3 ($TO_{23}$) were identified. These coupling transfer functions are given by

$$TO_{13}(s) = -\frac{643.383}{s^2 + 18s + 8100} = -\frac{643.83}{s + 9 \pm j89.5} \tag{5.18}$$

$$TO_{23}(s) = -\frac{0.9576\,(s^2 + 70s + 2500)}{s^2 + 119s + 7225} = -\frac{0.9576\,(s + 35 \pm j35.707)}{s + 59.5 \pm j60.702} \tag{5.19}$$

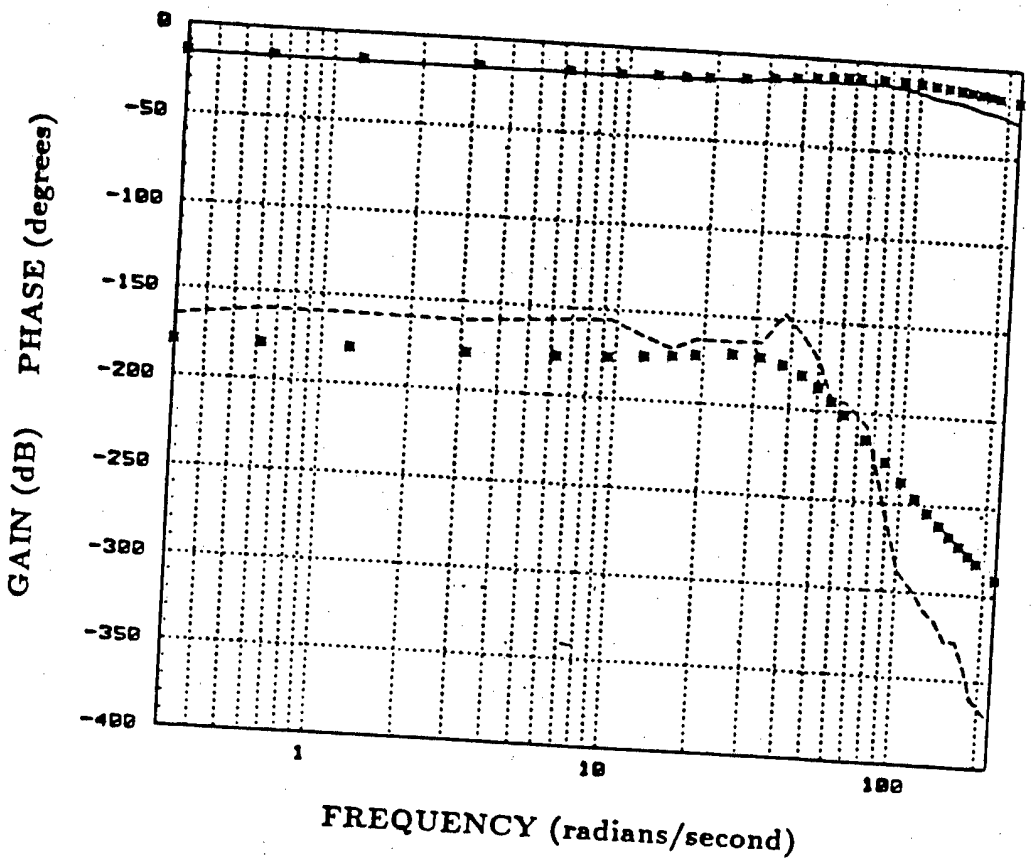



——— gain plot obtained from experimental data

- - - phase plot obtained from experimental data

* * * gain and phase plots of the estimated model

**Figure 5.19 Gain and phase plots of output-output coupling between joints 1 and 2 and those of their estimated model**




——— gain plot obtained from experimental data

- - - phase plot obtained from experimental data

* * * gain and phase plots of the estimated model

**Figure 5.20 Gain and phase plots of input-output coupling between joints 1 and 2 and those of their estimated model**

The gain-phase plots of the original data of coupling between joints 1 and 2 and those of the transfer function $TO_{12}(s)$ are shown in figures 5.19. The gain and phase plots of the other coupling transfer functions are given in Appendix C. It can be seen from these figures that the fit is not very good for the high frequency region. This was ignored, because the frequency range of interest is upto a maximum of 30 radians/sec. There is an initial shift of -180 degrees in all the coupling transfer functions. This was accounted for by introducing a negative sign in the transfer functions.

## 5.8 COMPARISON

The gain and phase plots of input-output coupling ($T_{xy}$ in figure 5.17) were obtained following the procedure given in the previous section. $K_p$ was set to 0.3 and $K_v$ was set to 0. The output-output coupling term ($TO_{xy}$ in figure 5.16) was multiplied with the the closed loop transfer function of joint x ($T_{xx}$ in figure 5.16) to obtain the coupling transfer function between input of joint x and output of joint y. The closed-loop transfer function was obtained using the same values of control parameters as given above. The gain and phase plots of this transfer function was compared with the gain and phase plots of input-output coupling obtained experimentally. Figure 5.20 shows the gain and phase plots of input-output coupling between joints 1 and 2 and of their estimated model. The gain and phase plots of input-output coupling between the other joints are given in Appendix C. It can be seen that the plots match closely. This comparison strengthens the validity of output-output coupling transfer functions that were identified experimentally.

After obtaining the transfer functions of coupling between different joints, an algorithm to decouple different joints of finger 1 of the DH was carried out as discussed in the next chapter.

# VI . DECOUPLING ALGORITHM

In this chapter, an alogithm to decouple different joints of finger 1 is discussed. Notations used in this chapter have the same explanations as those given in Chapter 5.

## 6.1 DECOUPLING ALGORITHM

The transfer functions of individual joints of finger 1 and those of the output-output coupling between different joints (figure 5.16) were identified in the previous chapter. The coupling between the input of joint x and the output of joint y ($T_{xy}$) is obtained by multiplying the closed-loop transfer function of joint x ($T_{xx}$ in figure 5.15) with the transfer function of the output-output coupling between joint x and joint y ($TO_{xy}$ in figure 5.15) as shown in figure 5.16. The coupling between joints x and y ($T_{xy}$) can be removed by adding to the input of joint y, the term $T_{yy}^{-1}T_{xy}$ as shown in figure 6.1.

The system shown in figures 5.15, 5.16 and 6.1 can be represented as

$$Y_1(s) = T_{11}(s)\,X_1(s) \tag{6.1}$$

$$Y_2(s) = T_{22}(s)\,X_2(s) + TO_{12}(s)\,X_1(s) \tag{6.2}$$

Substituting equation ( 6.1) in equation ( 6.2), we get

$$Y_2(s) = T_{22}(s)\,X_2(s) + T_{12}(s)\,X_1(s) \tag{6.3}$$

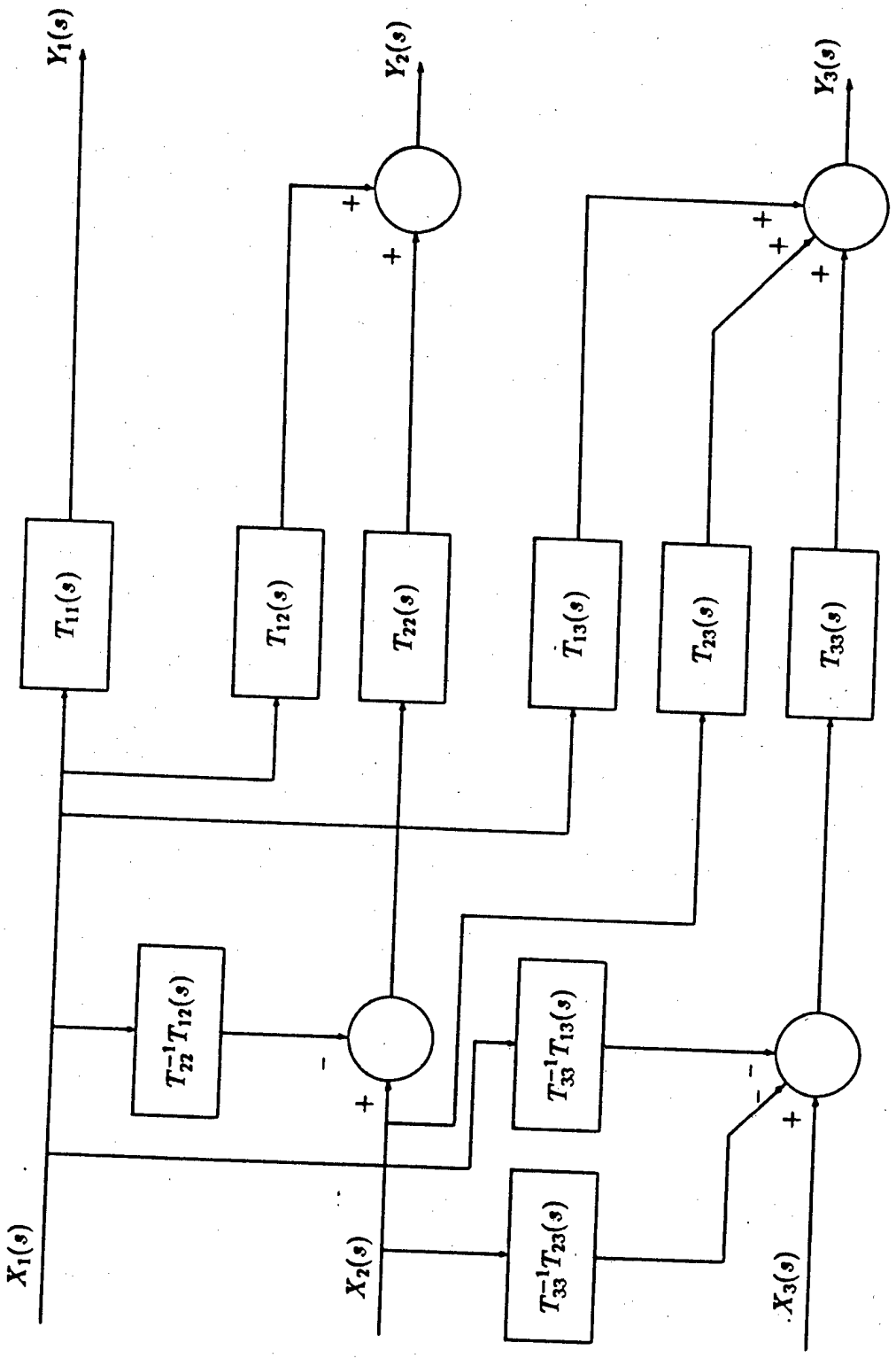



**Figure 6.1 Block diagram illustrating decoupling scheme**

where

$$T_{12}(s) = T_{11}(s)\,TO_{12}(s)$$

It can be seen from equation( 6.2) that the output of joint 2, $Y_2(s)$, is dependent upon the input to joints 1 and 2, $X_1(s)$ and $X_2(s)$, respectively. To remove the coupling between joint 1 and joint 2, i.e. the dependancy of $Y_2(s)$ on $X_1(s)$, the system can be modified by changing the input function of joint 2. This can be done by subtracting $T_{22}^{-1}(s)\,T_{12}(s)\,X_1(s)$ from the input function, $X_2(s)$, of joint 2. This makes the output $Y_2(s)$ independent of $X_1(s)$ and can be represented as

$$Y_2(s) = T_{12}(s)\,X_1(s) + T_{22}(s)\,\{X_2(s) - T_{22}^{-1}(s)\,T_{12}(s)X_1(s)\} \tag{6.4}$$

After cancellation, the equation becomes

$$Y_2(s) = T_{22}(s)\,X_2(s) \tag{6.5}$$

Joint 3 is decoupled from the other joints in a similar fashion. The input of joint 3 was modified to remove coupling between joint 3 and other joints. The decoupling equations of joint 3 are given by

$$Y_3(s) = T_{11}(s)TO_{13}(s)X_1(s) + T_{22}(s)TO_{23}(s)X_2(s) + T_{33}(s)X_3(s) \tag{6.6}$$

$$Y_3(s) = T_{13}(s)X_1(s) + T_{23}(s)X_2(s) + T_{33}(s)X_3(s) \tag{6.7}$$

where

$$T_{13}(s) = T_{11}(s)TO_{13}(s) \tag{6.8}$$

$$T_{23}(s) = T_{22}(s)TO_{23}(s) \tag{6.9}$$

$$Y_3(s) = T_{13}(s)X_1(s) + T_{23}(s)X_2(s) +$$

$$T_{33}(s)\{\, X_3(s) \;-\; T_{33}^{-1}(s)T_{13}(s)X_1(s) \;-\; T_{33}^{-1}(s)T_{23}(s)X_2(s)\} \quad (6.10)$$

$$= \; T_{33}(s)X_3(s) \quad (6.11)$$

## 6.2 REDUCED-ORDER DECOUPLING SCHEME

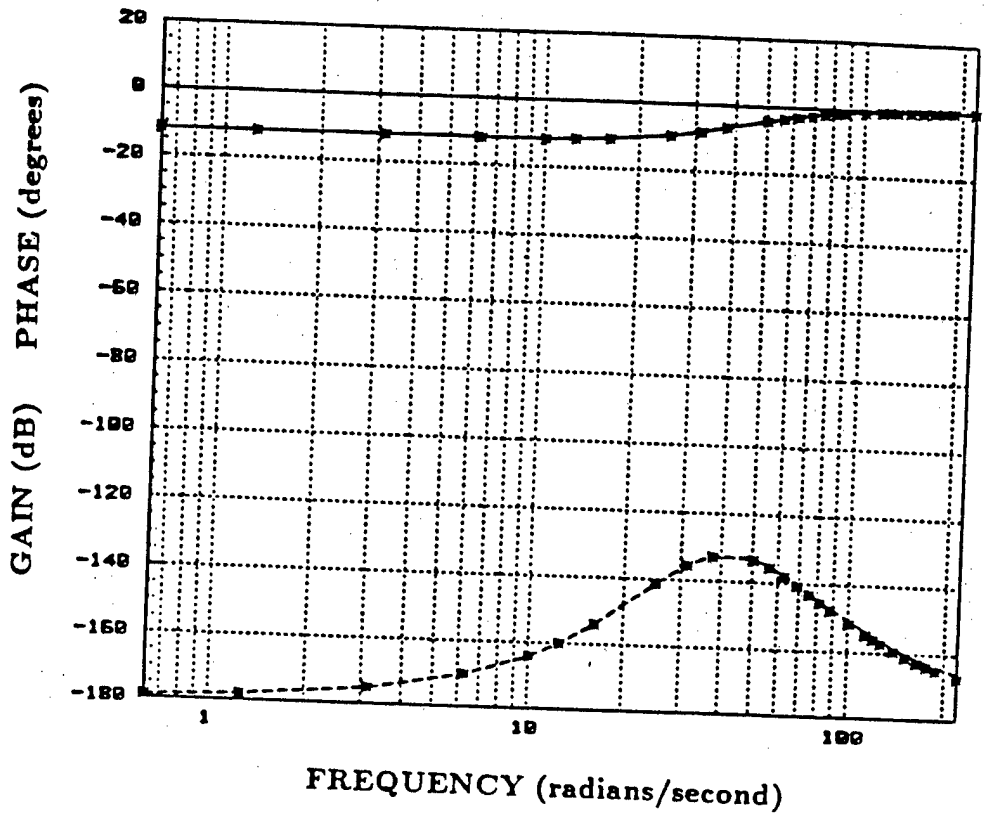
The closed-loop transfer functions of individual joints ($T_{xx}$ in figure 6.1) were derived from their respective open-loop transfer functions, using the design parameters given in Table 5.1. The coupling terms ($T_{xy}$ in figure 6.1) and the decoupling terms ($T_{yy}^{-1}T_{xy}$ in figure 6.1) were calculated. The decoupling terms resulted in fourth order equations which are given below

$$T_{22}^{-1}(s)T_{12}(s) \;=\; -\frac{(1.0829)\,(s+24\pm j18)\,(s+268.76)\,(s+317.26)}{(s+48\pm j36)\,(s+278.46)\,(s+322.12)} \quad (6.12)$$

It can be seen from equation ( 6.12) that the zeros -268.76 and -317.26 and poles -278.46 and -322.12 are far away from the imaginary axis in the left half plane when compared to the quadratic zeros and poles respectively. Their effect is negligible and hence can be ignored. Also, they do not contribute to the frequency reponse plot in the frequency range of interest (0.1 – 7.0 hertz). It is also evident from equation ( 6.12) that the simple poles and zeros are very close to each other and hence can be cancelled. Equation( 6.12) is thus reduced to a second-order transfer function given by

$$T_{22}^{-1}T_{12}(s) \;=\; (-1.0297)\,\frac{(s+24\pm j18)}{(s+48\pm j36)} \quad (6.13)$$

Note that the reduced-order transfer function is same as the output-output coupling transfer function $TO_{12}$ (equation 5.17) except for a gain factor and that all the

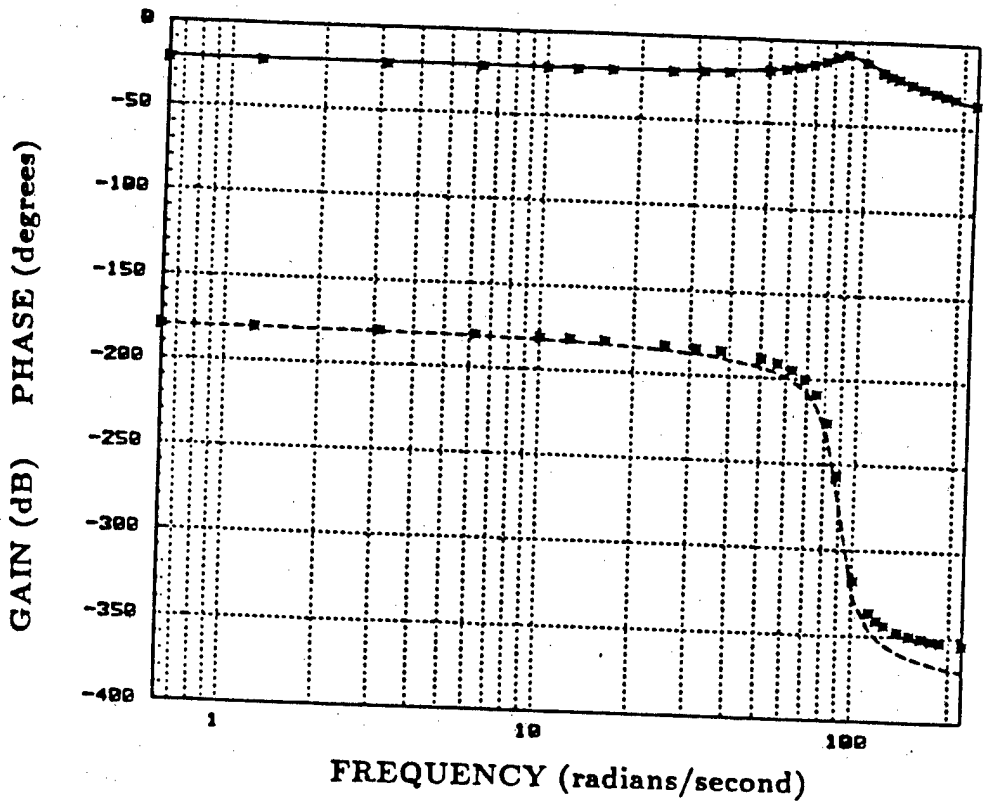



——— gain plot of the unreduced coupling transfer function between input of joint 1 and output of joint 2

- - - phase plot of the unreduced coupling transfer function between input of joint 1 and output of joint 2

* * * gain and phase plots of coupling transfer functions between the outputs of joints 1 and 2

**Figure 6.2 Comparison of gain and phase plots of decoupling terms with their reduced-order models**

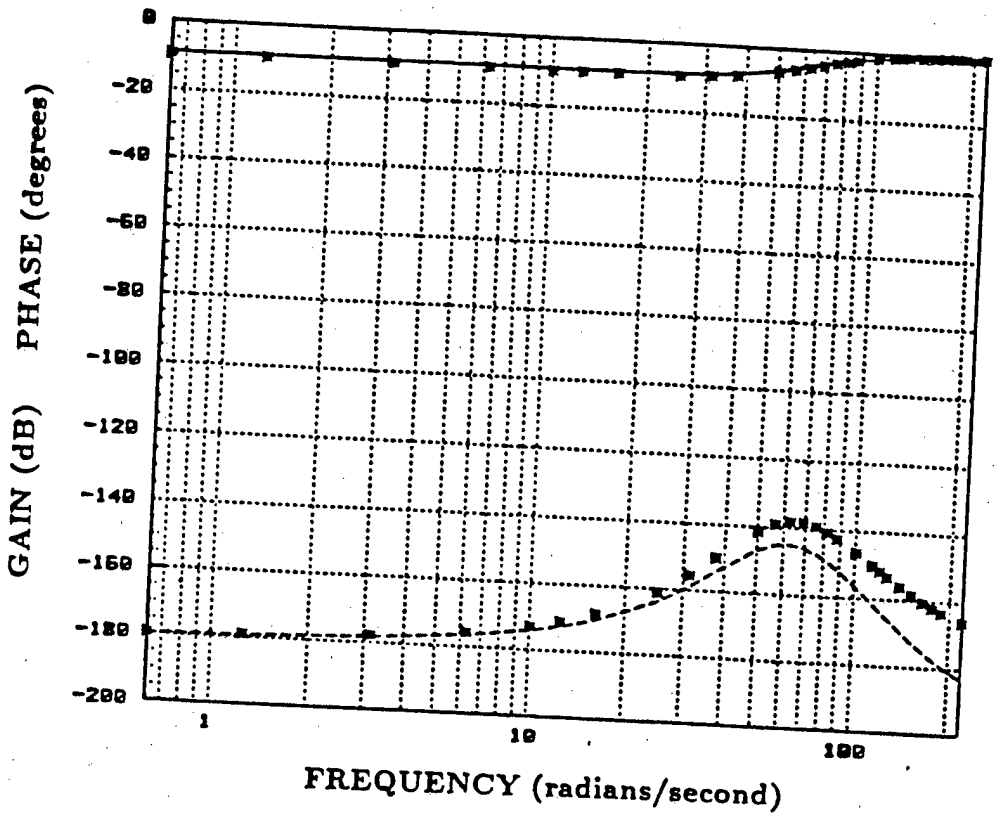



——— gain plot of the unreduced coupling transfer function between input of joint 1 and output of joint 3

- - - phase plot of the unreduced coupling transfer function between input of joint 1 and output of joint 2

* * * gain and phase plots of coupling transfer functions between the outputs of joints 1 and 3

**Figure 6.3 Comparison of gain and phase plots of decoupling terms with their reduced-order models**




——— gain plot of the unreduced coupling transfer function between input of joint 2 and output of joint 3

- - - phase plot of the unreduced coupling transfer function between input of joint 2 and output of joint 3

* * * gain and phase plots of coupling transfer functions between the outputs of joints 2 and 3

**Figure 6.4 Comparison of gain and phase plots of decoupling terms with their reduced-order models**

discarded zeros and poles are the poles of the closed loop transfer functions of joints 1 and 2 . This is because joints 1 and 2 have high bandwidths, which are increased further by choosing a high value of $K_p$ (equation 5.14).

In the same way, the decoupling terms $T_{33}^{-1}T_{13}$ and $T_{33}^{-1}T_{23}$ are reduced to second-order transfer functions which are same as $TO_{13}$ (equation 5.17) and $TO_{23}$ (equation 5.18) respectively except for a gain factor. The decoupling terms $T_{33}^{-1}T_{13}$ and $T_{33}^{-1}T_{23}$ and their reduced versions are given below

$$T_{33}^{-1}(s)T_{13}(s) = -\frac{(389.95)\,(s+390.48)\,(s+422.0538)}{(s+9\pm j89.5)\,(s+278.47)\,(s+322.12)} \tag{6.14}$$

$$= (-1.113)\,\frac{643.383}{s+9\pm j89.5} \tag{6.15}$$

$$T_{33}^{-1}(s)T_{23}(s) = (1.0818)\,\frac{(.9576)(s+35\pm j35.707)(s+390.48)(s+422.0538)}{(s+59.5\pm j60.7)\,(s+268.76)\,(s+317.27)} \tag{6.16}$$

$$= (-1.0818)\,\frac{(.9576)\,(s+35\pm j35.707)}{(s+59.5\pm j60.702)} \tag{6.17}$$

To check the validity of the reduced-order model, a comparison of the gain and phase plots of $T_{22}^{-1}T_{12}$ and $TO_{12}$ (figure 6.2), the gain and phase plots of $T_{33}^{-1}T_{13}$ and $TO_{13}$ (figure 6.3) and those of $T_{33}^{-1}T_{23}$ and $TO_{23}$ (figure 6.4) was performed. It can be seen that the gain and phase plots match very well. Difference equations for the decoupling terms were computed and a C code was written to implement the difference equations (given in Appendix E).

## 6.3 SIMULATION

The simulation of the different joints of finger 1 and the decoupling algorithm was performed using SYSTEM_BUILD in MATRIX$_x$, a powerful computer aided design

package.

### 6.3.1 MATRIXx

$MATRIX_x$ is a powerful programmable, matrix calculator with excellent graphical capabilities. With this, it is possible to solve complex, large-scale matrix problems.

### 6.3.2 SYSTEM_BUILD

SYSTEM_BUILD is a utility of $MATRIX_x$ which provides an interactive, menu-driven graphical environment for building, modifying and editing complex computer simulation models. Simulating sytem performance under both nominal and strained environment is easily accomplished with SYSTEM_BUILD. SYSTEM_BUILD also provides modularity in design which makes testing, modifying and interfacing parts of a model, a simple task.

### 6.3.3 BASIC BUILDING UNIT - A BLOCK

SYSTEM_BUILD basic building unit is the block. SYSTEM_BUILD has a large library of different types of blocks. The main block categories are

1. SUPER-BLOCK
2. GAIN BLOCK
3. PIECE-WISE LINEAR FUNCTIONS
4. DYNAMIC BLOCKS

5. TRIGONOMETRIC FUNCTIONS

6. USER CODE BLOCKS

7. COORDINATE TRANSFORMATIONS

8. SIGNAL SOURCES

9. LOGICAL FUNCTIONS

10. ALGEBRAIC EQUATIONS

11. POWER-EXPONENTIAL-LOGARITHM

Figure 6.5 shows the general architecture of a block.

When using SYSTEM_BUILD, the system prompts the user for the type of block, block name, number of inputs, number of outputs, number of states and block parameters. The next step is to determine the simulation time duration and the input forcing functions. Once that has been done, the model is ready to run.

### 6.3.4 SUPER-BLOCK

A very important type of block is SUPER-BLOCK. This block can have upto one hundred other blocks interfaced in any way. The most important advantage of a SUPER-BLOCK is that it can contain other SUPER-BLOCKS. This permits the nesting of SUPER-BLOCKS as seen in figure 6.6. This nesting property makes it possible to implement complex systems which contain any number of blocks.

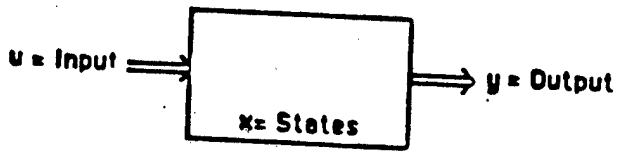



Figure 6.5 General architecture of a block

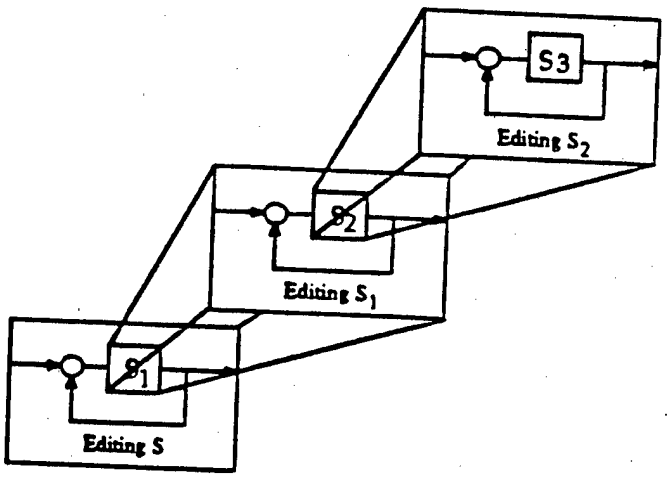



Figure 6.6 Nesting of super-blocks

## 6.4 SIMULATION OF THE DECOUPLING ALGORITHM

The decoupling algorithm as explained in section 6.2 was implemented as shown in figure 6.7. Block1, Block2, Block3 and Block4 are implemented as super-blocks in $MATRIX_x$. $T_{xx}$ shown in figure 6.1, represents the closed loop control of joint x and is implemented as a super-block Txx. Figure 6.8 represents the servo loop of joint x ($T_{xx}$) with the feedback controller and the series controller, which is collectively called the PID controller. Block1 simulates the closed loop control of joint 1, Block2 represents the simulation of joint 2 and that of coupling of other joints with joint 2. Block3 and Block4 simulate the closed loop control of joint 3 and the coupling between joint 3 and joints 1 and 2. The four super-blocks Block1, Block2, Block3 and Block4 are incorporated in another super-block named DEC (shown in figure 6.7). Super-Block DEC represents the entire system (joints 1, 2 and 3 of finger 1). The block diagrams which represent the system as implemented in $MATRIX_x$, is given in Appendix G.

## 6.5 SIMULATION RESULTS

To illustrate the coupling between joints 1, 2 and 3 of finger 1, the system shown in figure 5.15 was simulated in $MATRIX_x$ (as explained in section 6.4). A sinusoidal signal was applied to the input of joint 1 ($X_1(s)$ in figure 5.15) and the responses of joints 1, 2 and 3 were recorded and plotted (figure 6.9). It can be seen from figure 6.9 that the outputs of joints 2 and 3 are affected due to the input at joint 1. Similarly, figure 6.10 illustrates the coupling between joints 2 and 3 due to input at joint 2.

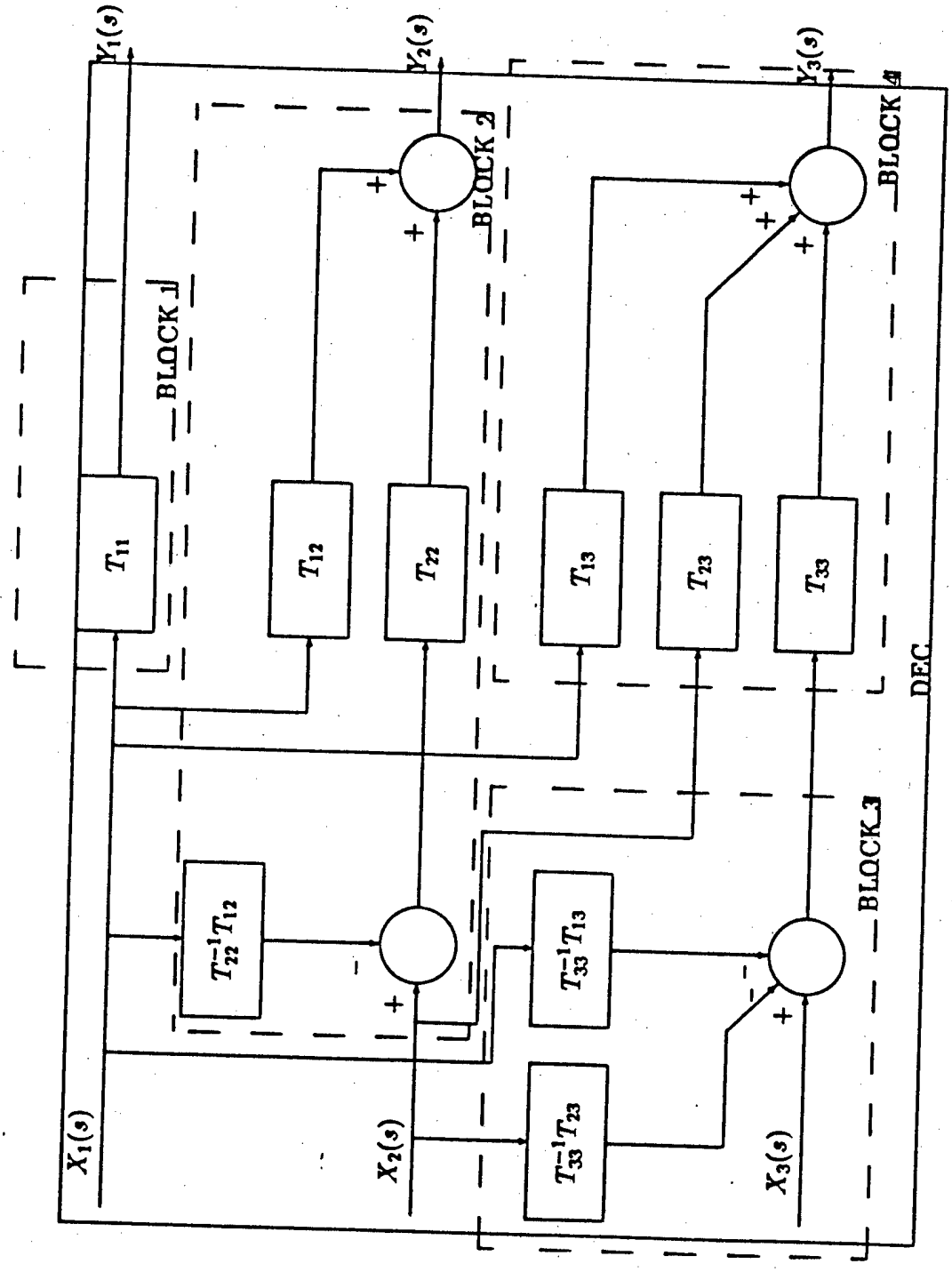



**Figure 6.7 Block diagram of the decoupling algorithm showing the different super-blocks**

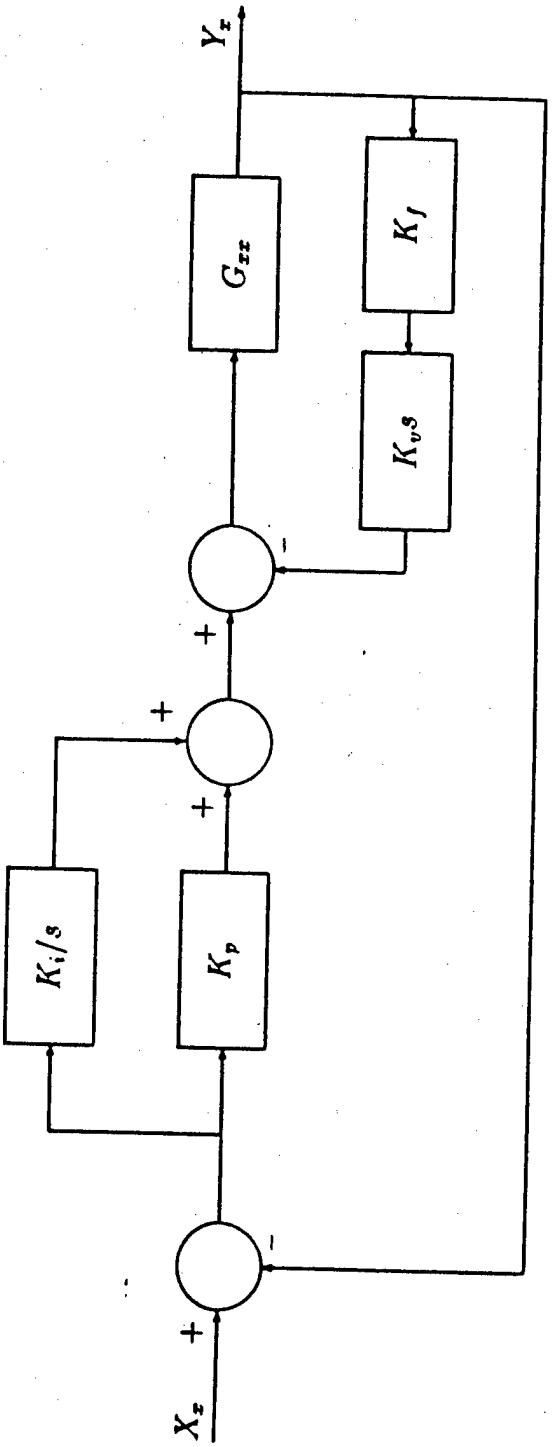
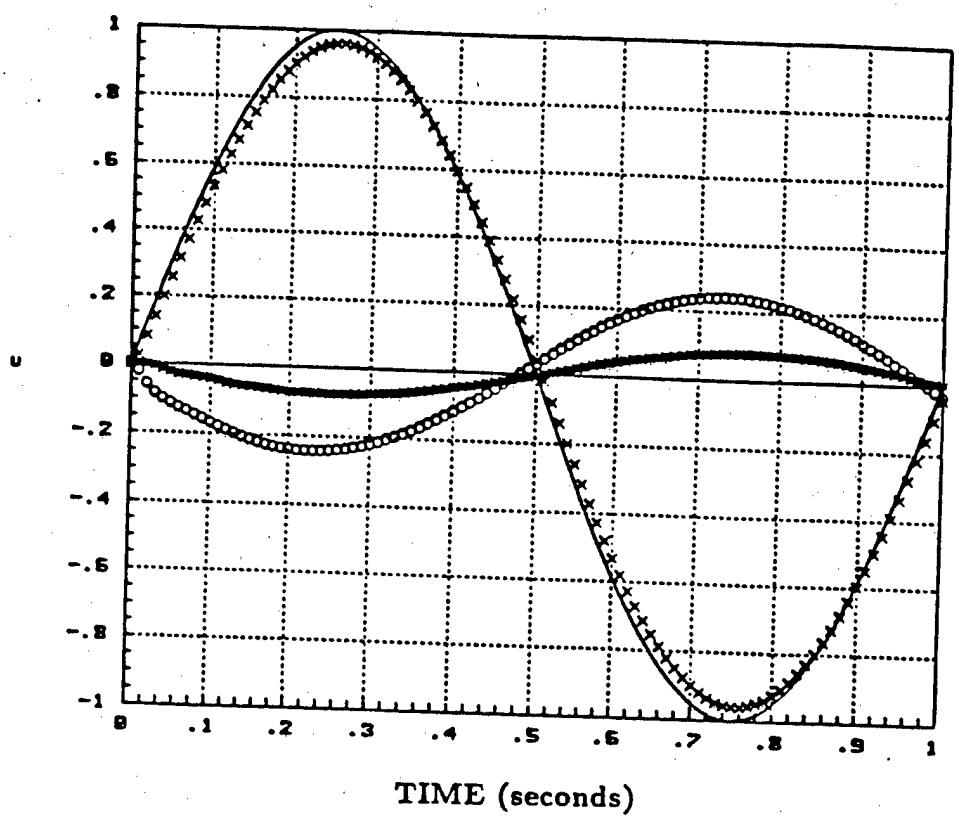



**Figure 6.8 Block diagram of closed loop position control of a single joint**



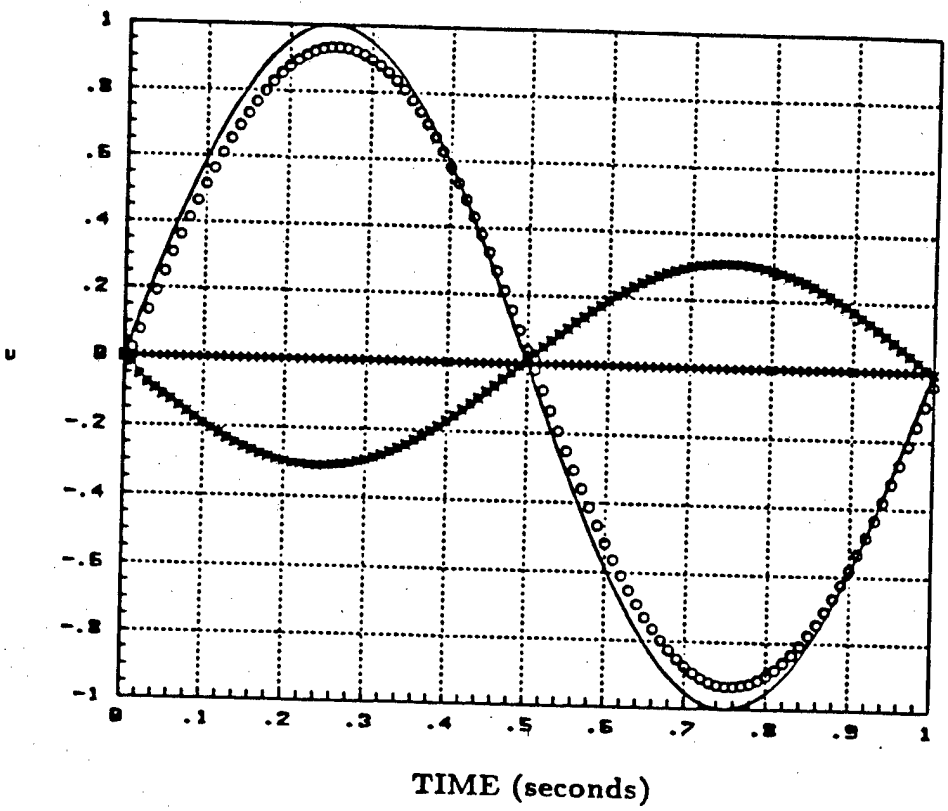

——— sinusoidal input of joint 1

xxxx response of joint 1

oooo response of joint 2

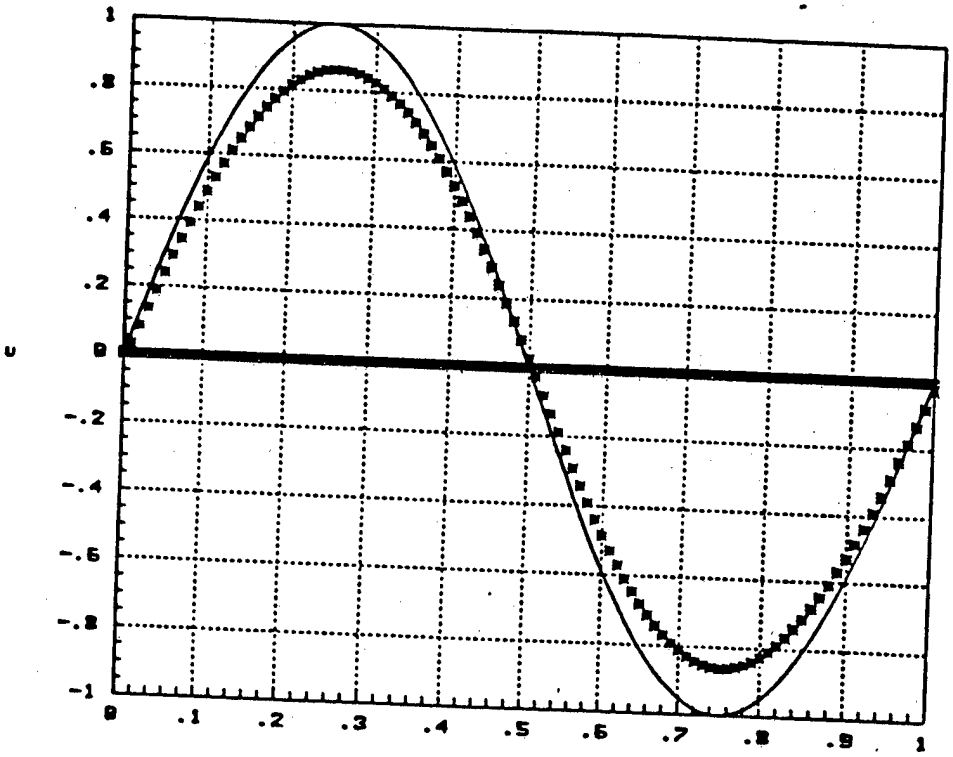
* * * response of joint 3

**Figure 6.9** **Response of joints 1, 2 and 3 for an input at joint 1 without the decoupling scheme**



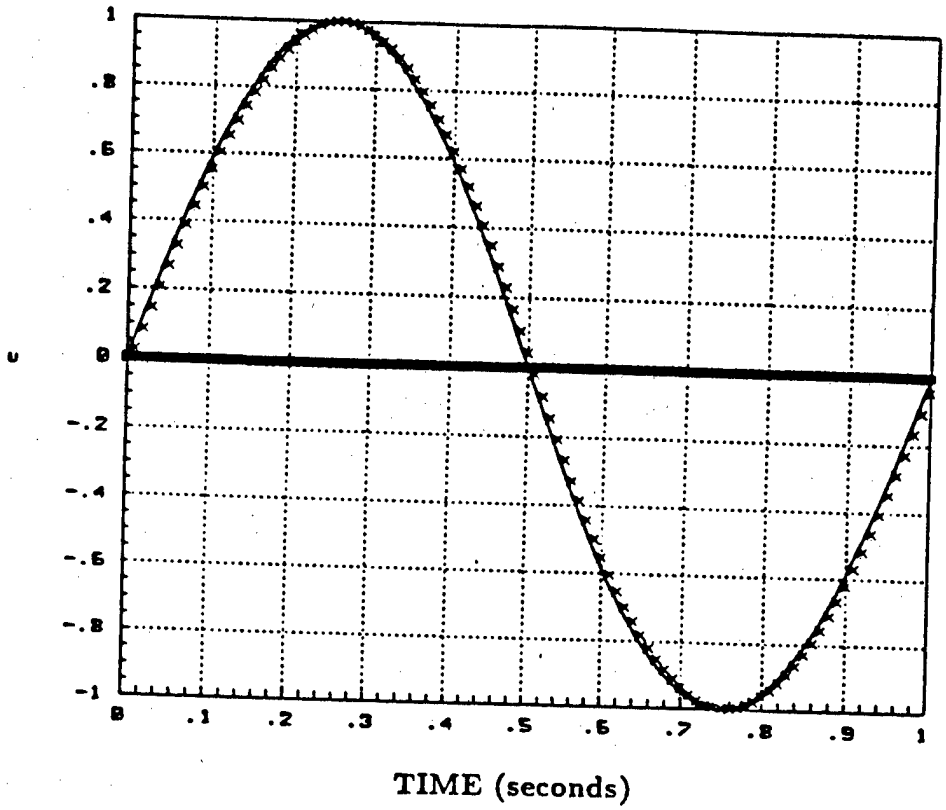

| | |
|---|---|
| —— | sinusoidal input of joint 2 |
| xxxx | response of joint 1 |
| oooo | response of joint 2 |
| * * * | response of joint 3 |

**Figure 6.10 Response of joints 1, 2 and 3 for an input at joint 2 without the decoupling scheme**




| | |
|---|---|
| —— | sinusoidal input of joint 3 |
| xxxx | response of joint 1 |
| oooo | response of joint 2 |
| * * * | response of joint 3 |

**Figure 6.11 Response of joints 1, 2 and 3 for an input at joint 3 without the decoupling scheme**



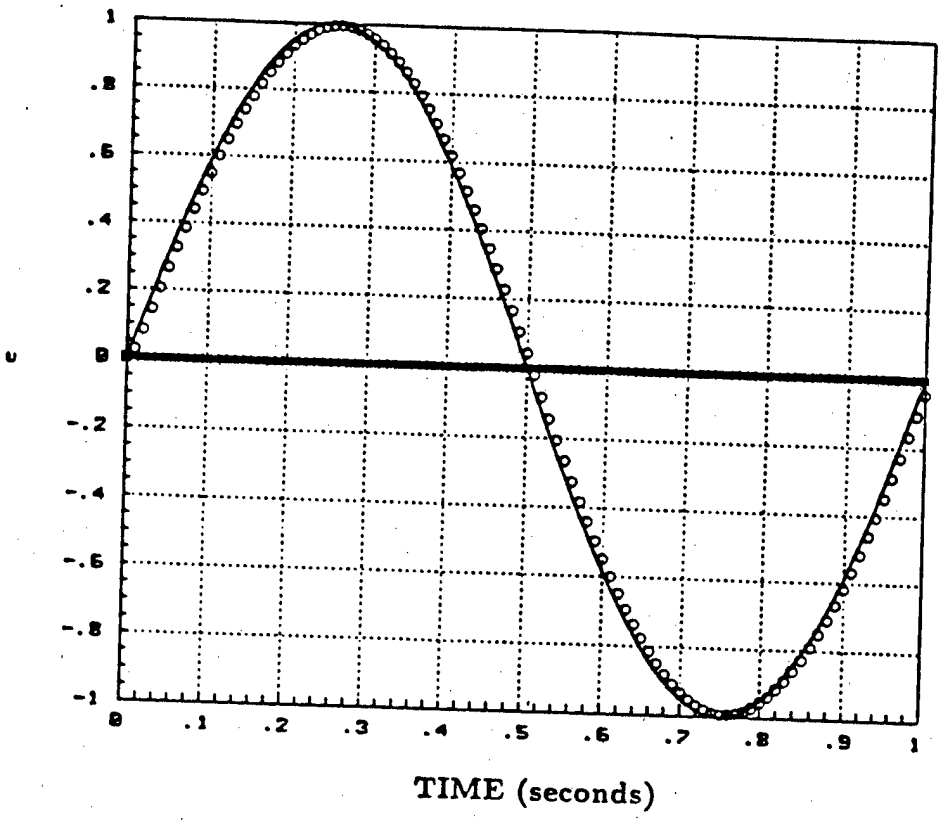

sinusoidal input of joint 1

xxxx response of joint 1

oooo response of joint 2

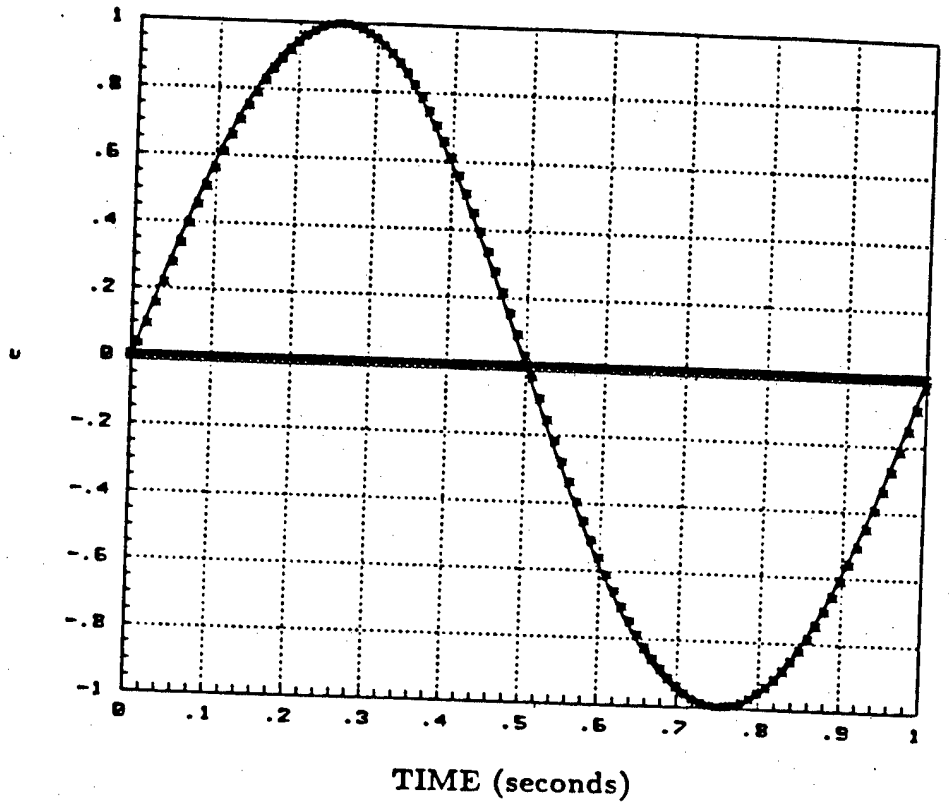
* * * response of joint 3

**Figure 6.12 Response of joints 1, 2 and 3 for an input at joint 1 with the decoupling scheme**




| | |
|---|---|
| —— | sinusoidal input of joint 2 |
| xxxx | response of joint 1 |
| oooo | response of joint 2 |
| * * * | response of joint 3 |

**Figure 6.13 Response of joints 1, 2 and 3 for an input at joint 1 with the decoupling scheme**




| | |
|---|---|
| —— | sinusoidal input of joint 3 |
| xxxx | response of joint 1 |
| oooo | response of joint 2 |
| * * * | response of joint 3 |

**Figure 6.14 Response of joints 1, 2 and 3 for an input at joint 1 with the decoupling scheme**

The responses of joints 1, 2 and 3 due to a sinusoidal input at joint 3 is shown in figure 6.11. It can be seen from figure 6.10 that there is no response at joint 1, similarly in figure 6.11, the responses of joints 1 and 2 are zero. This is because the coupling exists between a joint commanded to move and the distal joints only, there is no coupling between a joint at which the input is applied and a proximal joint (Table 5.2). The decoupling scheme (figure 6.1) was simulated, the results of which are shown in figures 6.12 – 6.14. It can be seen that the coupling has been removed completely. When an input was applied to joint x, there was output at joint x only, the response of other joints were zero. Note that figures 6.12 – 6.14 show the response of joints 1, 2 and 3 with the addition of $K_g$ (figure 5.13).

Thus the simulation of the decoupling algorithm was performed successfully. The conclusion of this report is given in the next chapter.

# VII . SUMMARY AND CONCLUSION

The goal of the Robotic Telepresence (R/T) program at the Harry G. Armstrong Aerospace Medical Research Laboratory (AAMRL) at WPAFB is to develop a series of dynamic telepresence test cells integrating state-of-the-art components such as visual displays, sensors and robotic manipulators. One of the objectives of the R/T program is to effectively slave the Utah/M.I.T. Dextrous Hand (DH) to an exoskeleton master worn by a human operator. This would enable the DH to be put to use in hazardous environments or deep-sea or space applications, being driven remotely by a human operator. Each joint of the human hand is mapped to that of the DH using master/slave transformations. Joints 1, 2 and 3 of the DH are coupled. To enable the control of each joint of the DH separately by the exoskeleton master, the joints need to be decoupled.

The objective of this thesis is to design a controller that would decouple the different joints of finger 1 of the DH. The different gain blocks in the analog controller and the position and tendon tension sensors were calibrated. The gains produced by the gain blocks were found to vary with the input voltage. The flexion and extension bias settings were also determined. The individual joint loop transfer functions were computed using the frequency response method, and were determined to be of type zero. There is a significant coupling between joints 1, 2 and 3. The coupling was identified and an algorithm to decouple the joints was developed. The decoupling

algorithm was simulated using the SYSTEM_BUILD utility in $MATRIX_x$. The simulation results show that the algorithm was able to remove coupling between the joints. Software code to implement this algorithm was written in 'C' programming language.

## 7.1 RECOMMENDATIONS

The following recommendations are made to improve the existing control system :

1. It was found that there is a steady-state error between the position output and input when a step input is applied to the joints. This is due to the fact that the transfer functions of the joints are of type zero. One way to remove this error is by adding an integral term to the existing PD controller, which would result in zero steady-state error to a step input. The error can also be reduced by introducing a gain factor outside the position loop of individual joints. (figure 6.25).

2. The gains produced by different gain blocks of the analog controller are dependent upon the magnitude of the inputs (Appendix I). Further, the control parameters $K_p$ and $K_v$ cannot be set to a value greater than 10. The analog controller can be replaced by a microprocessor-based digital controller. This would enable the user to set precise gain factors and set $K_p$ and $K_v$ to larger values.

3. A further study should be made to determine the reasons for the coupling between joints and also the effect of changes in cocontraction on the transfer

functions of individual joints.

# APPENDICES

# APPENDIX A

# NOTATIONS

| | | |
|---|---|---|
| FxJy | – | finger x joint y , $0 \leq x \leq 3$,$0 \leq y \leq 3$ |
| GPVCALxy | – | Static calibration data file of position and velocity gain control blocks, finger x joint y |
| GEFCALxy | – | Static calibration data file of extension and flexion gain blocks of finger x joint y |
| PxyCAL | – | Static calibration data file of the position Hall effect sensors of finger x joint y |
| TxyFCAL | – | Calibration data file of the flexion tendon tension sensor of finger x joint y |
| TxyECAL | – | Calibration data file of the extension tendon tension sensor of finger x joint y |
| PD | – | Position desired |
| POS | – | Position output of the Hall effect sensor |
| EXT | – | Output of the extensor tendon tension sensor |
| FLX | – | Output of the flexor tendon tension sensor |
| GP | – | Gain setting of the position gain block |
| GV | – | Gain setting of the velocity gain block |

| | | |
|---|---|---|
| GE | – | Gain setting of the extension gain block |
| GF | – | Gain setting of the flexion gain block |
| BE | – | Extension bias setting |
| BF | – | Flexion bias setting |
| $G_{xx}(s)$ | – | Open loop transfer function of finger x |
| $T_{xx}(s)$ | – | Closed loop transfer function of finger x joint x |
| $TO_{xy}(s)$ | – | transfer function of coupling between the output of joint x and output of joint y due to an input at joint x |
| $T_{xy}(s)$ | – | transfer function of coupling between input of joint x and output of joint y due to an input at joint x |
| $X_x(s)$ | – | input at joint x |
| $Y_x(s)$ | – | output at joint x |
| $K_p$ | – | proportional constant of the PID controller |
| $K_v$ | – | differential constant of the PID controller |
| $K_f$ | – | Gain factor of the differentiator circuit |
| $K_i$ | – | integral constant of the PID controller |

# APPENDIX B

# CALIBRATION TABLES AND PLOTS

This section gives the tables and plots of the calibration of

1. Position, flexion and extension gain blocks of joints 0, 1 2, and 3 of finger 1

2. Position sensors of joints 0, 1, 2 and 3 of finger 1

3. Flexion and extension tendon tension sensors of joints 0, 1, 2 and 3 of finger 1

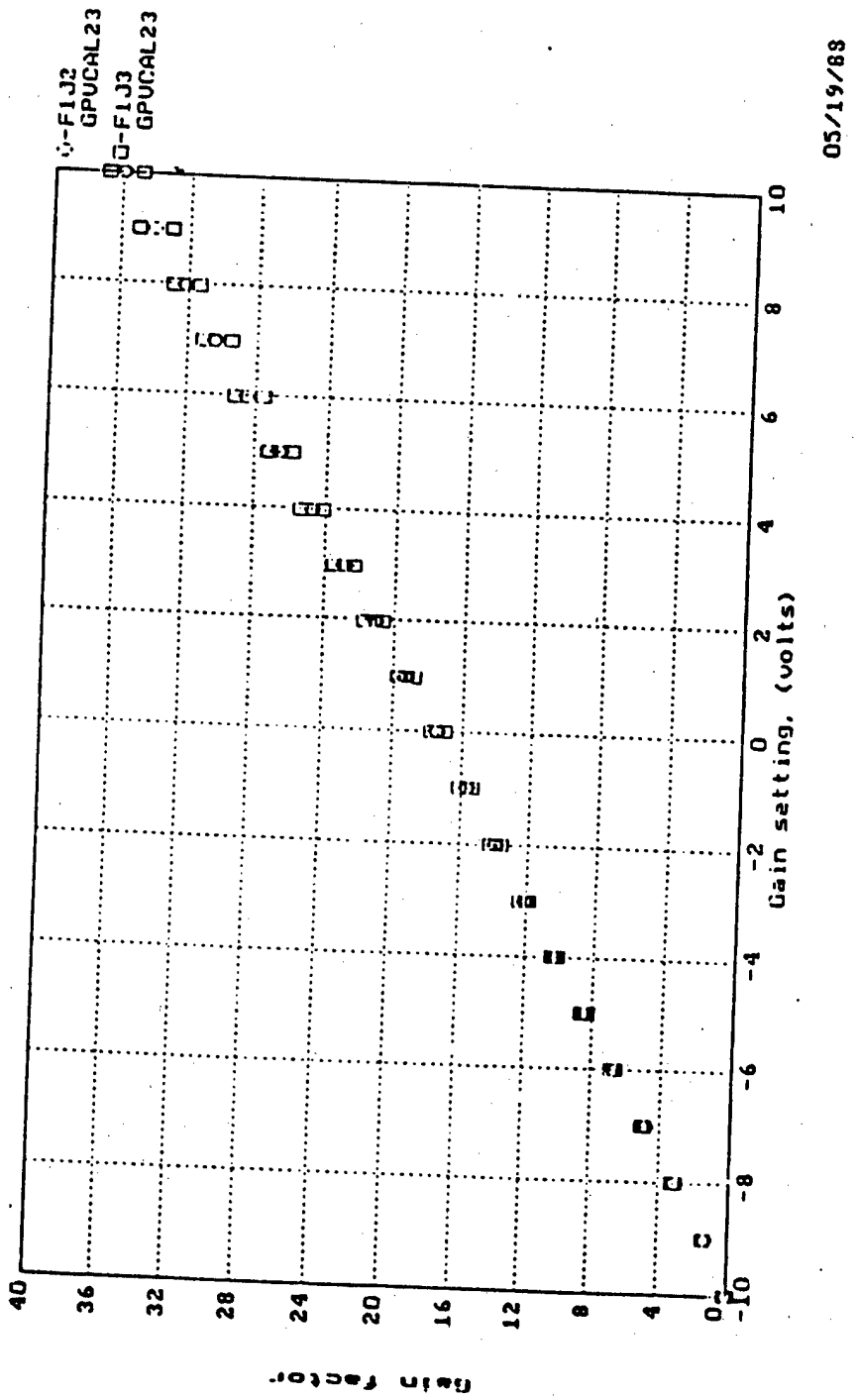



Figure B1 Position Gain Block Calibration

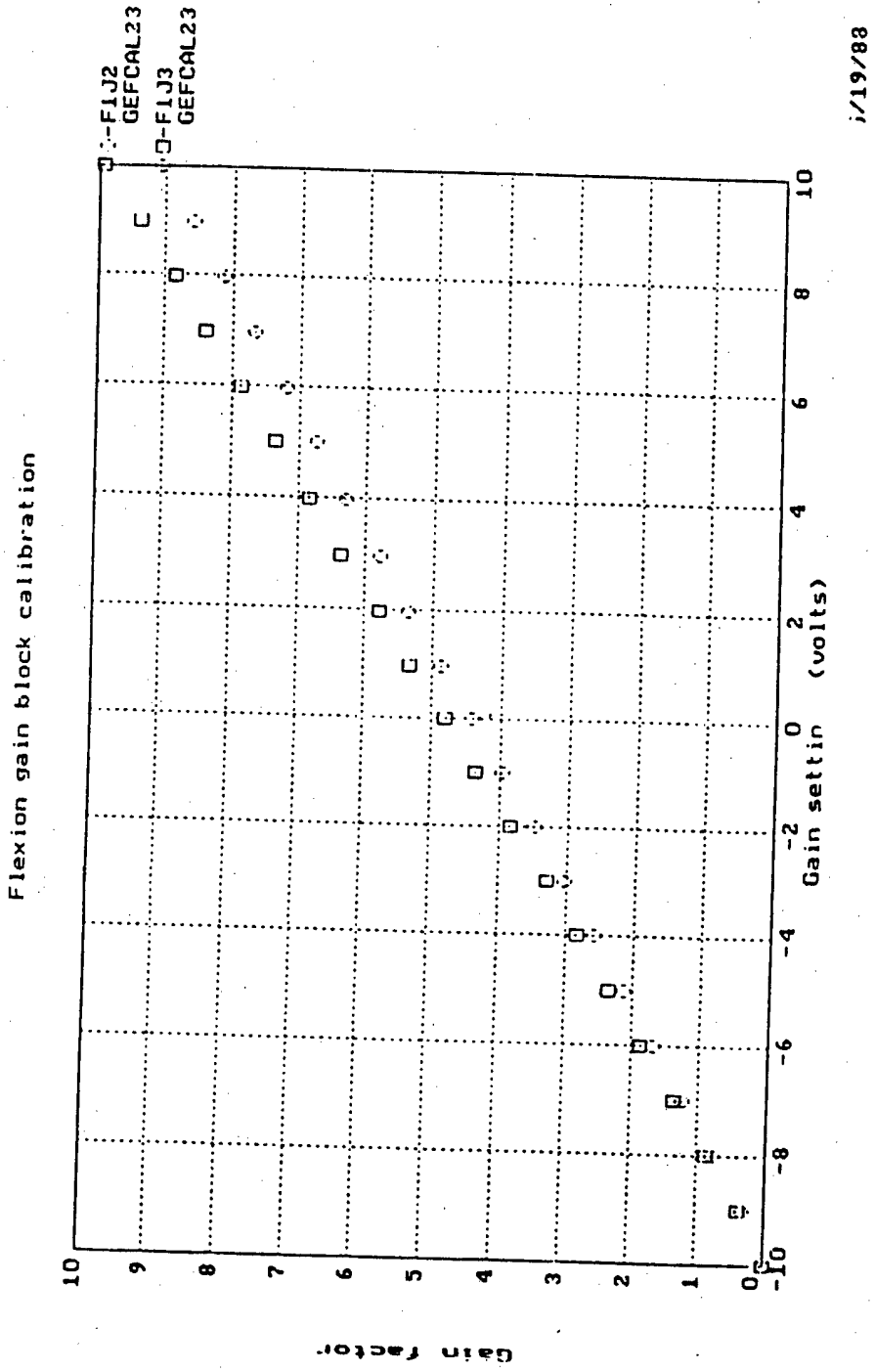



Figure B2 Flexion Gain Block Calibration

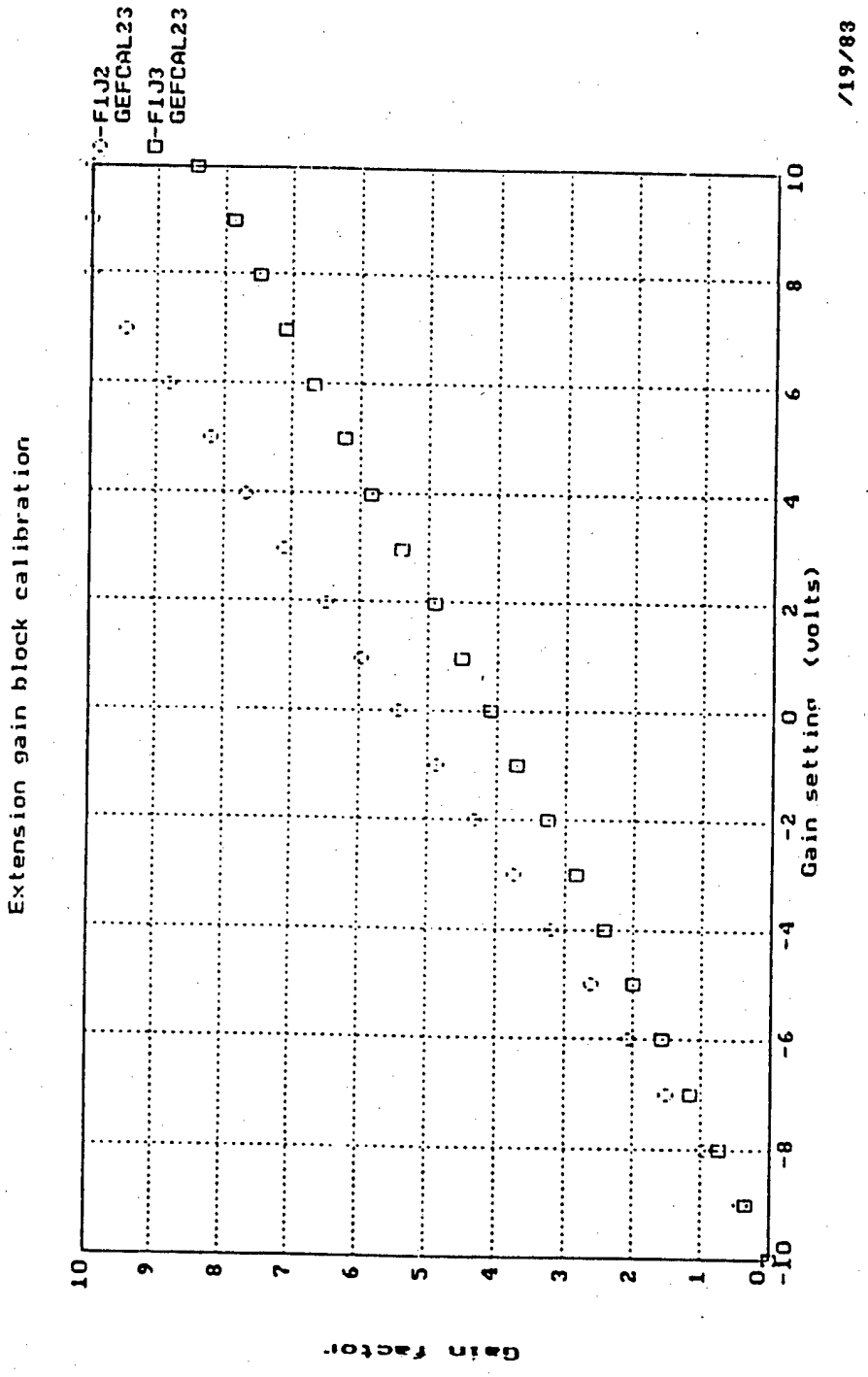



**Figure B3 Extension Gain Block Calibration**

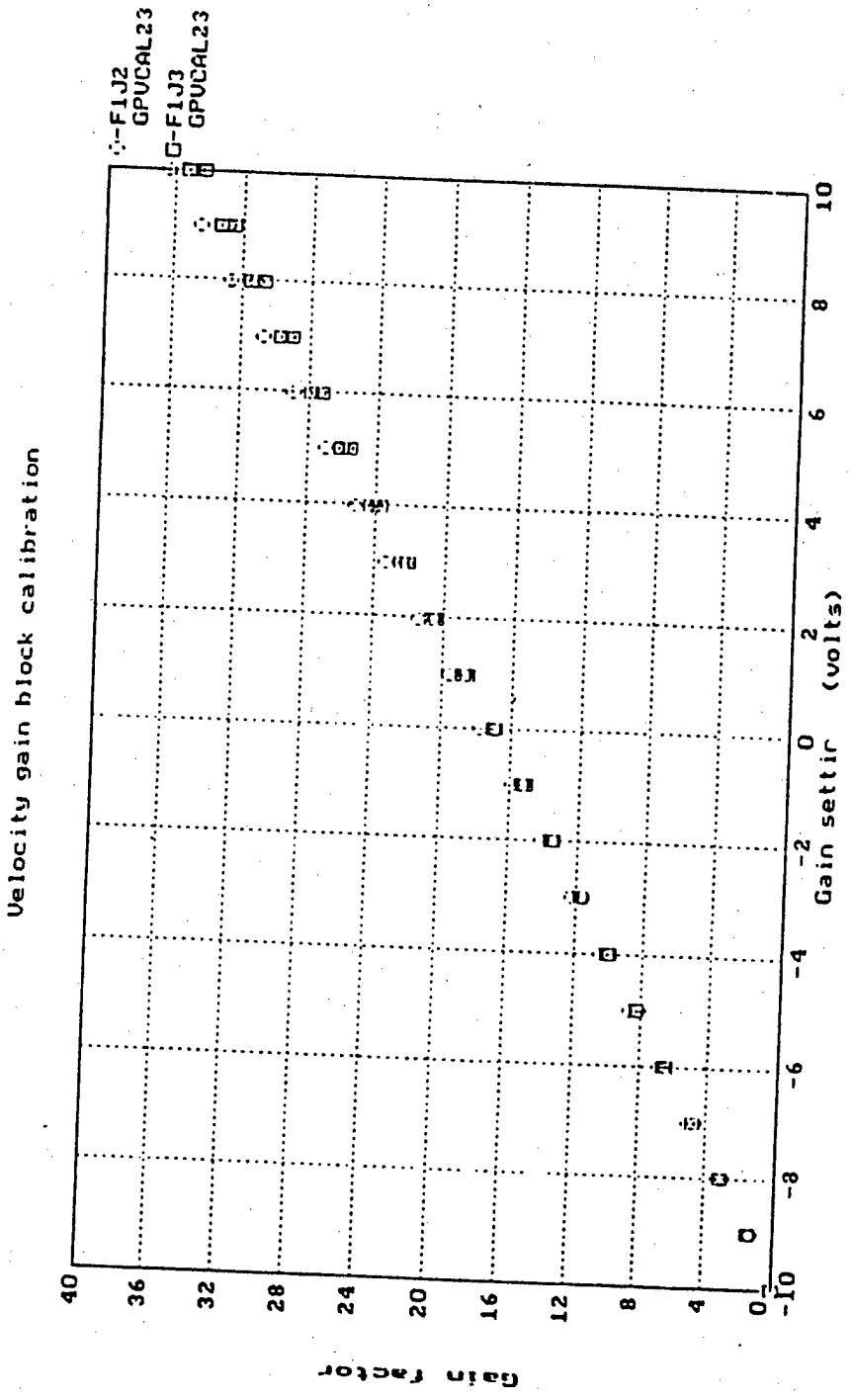



Figure B4 Velocity Gain Block Calibration

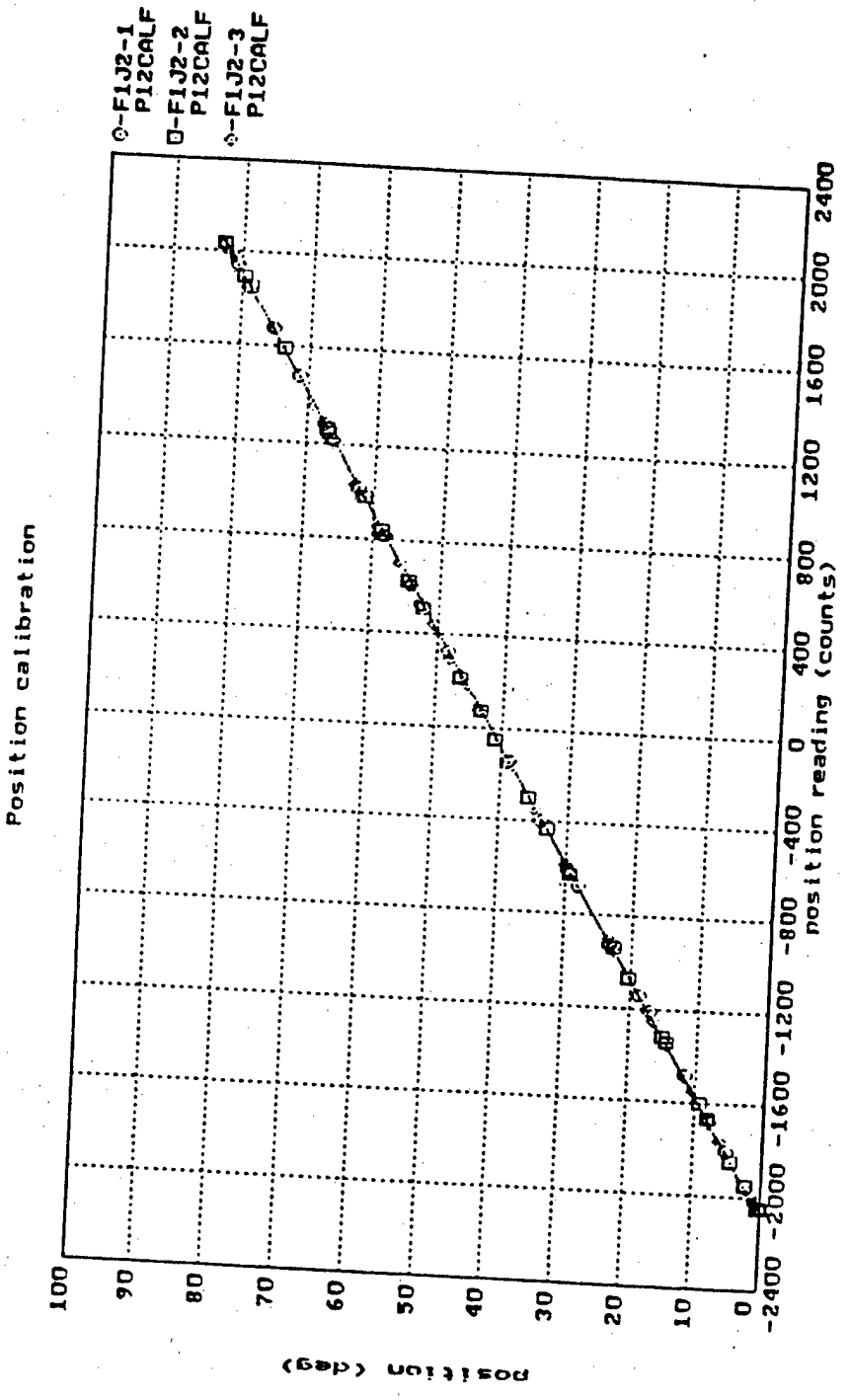



Figure B5 Position Sensor Calibration

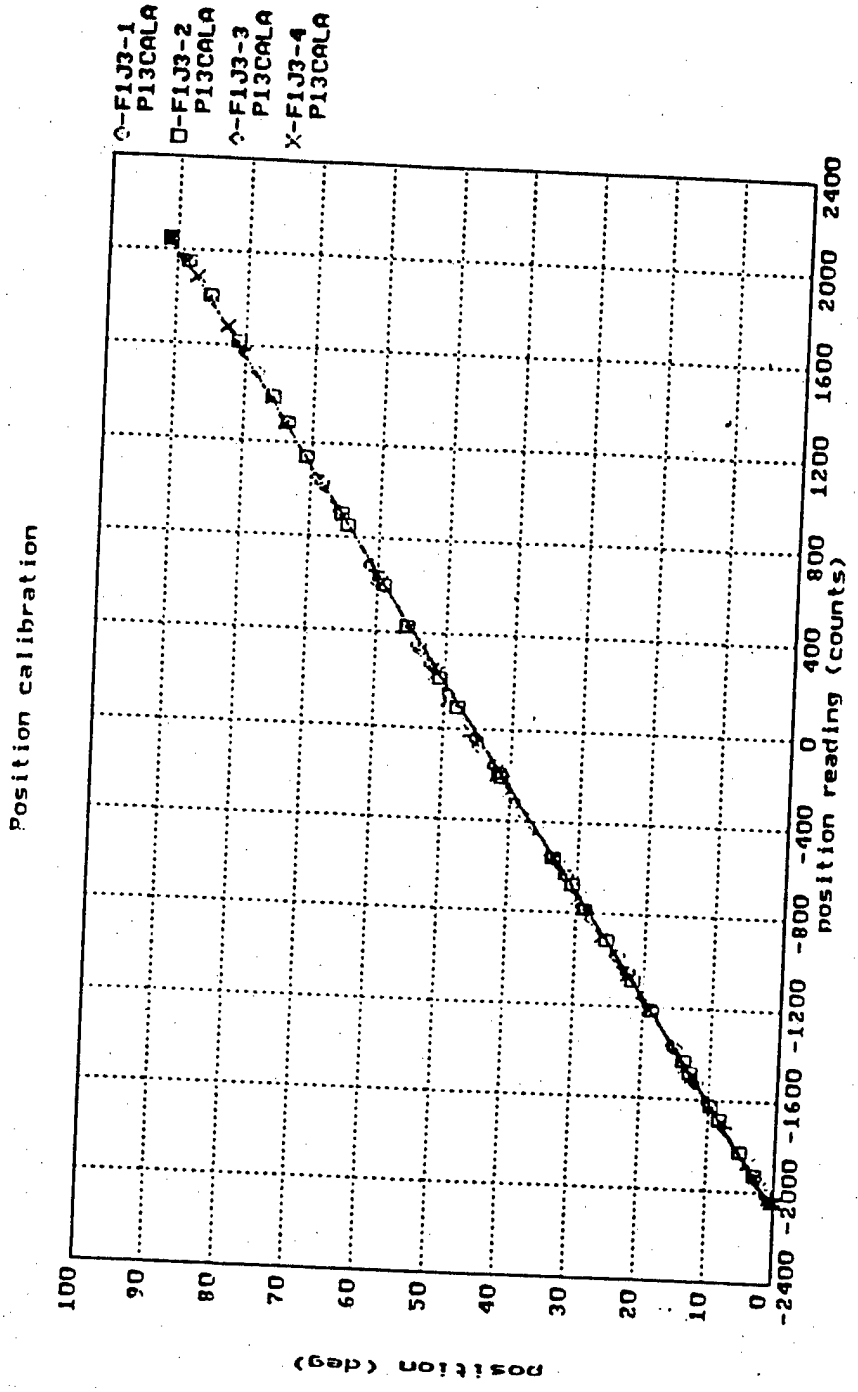



Figure B6 Position Sensor Calibration

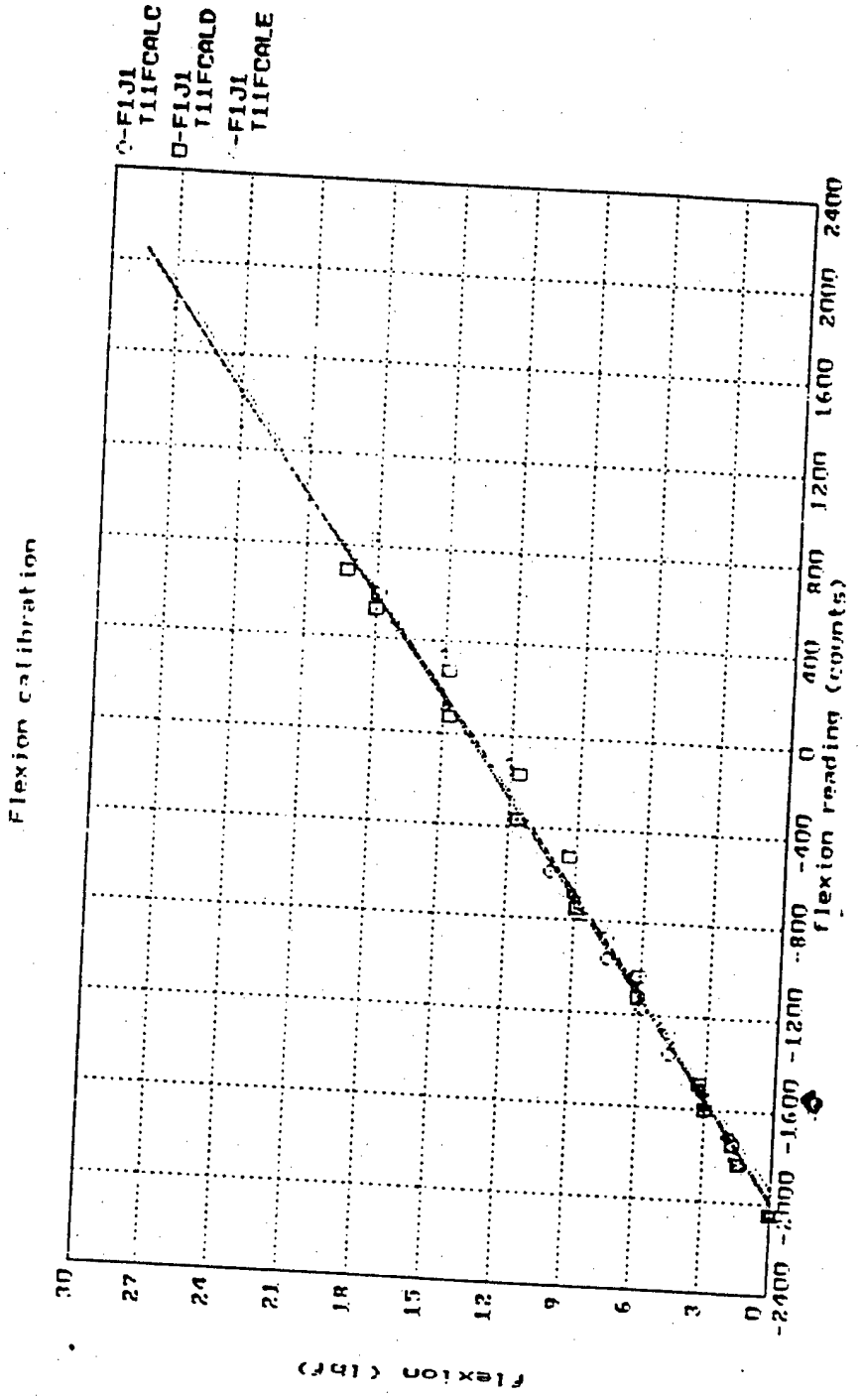



Figure B7 Flexion Tendon Sensor Calibration

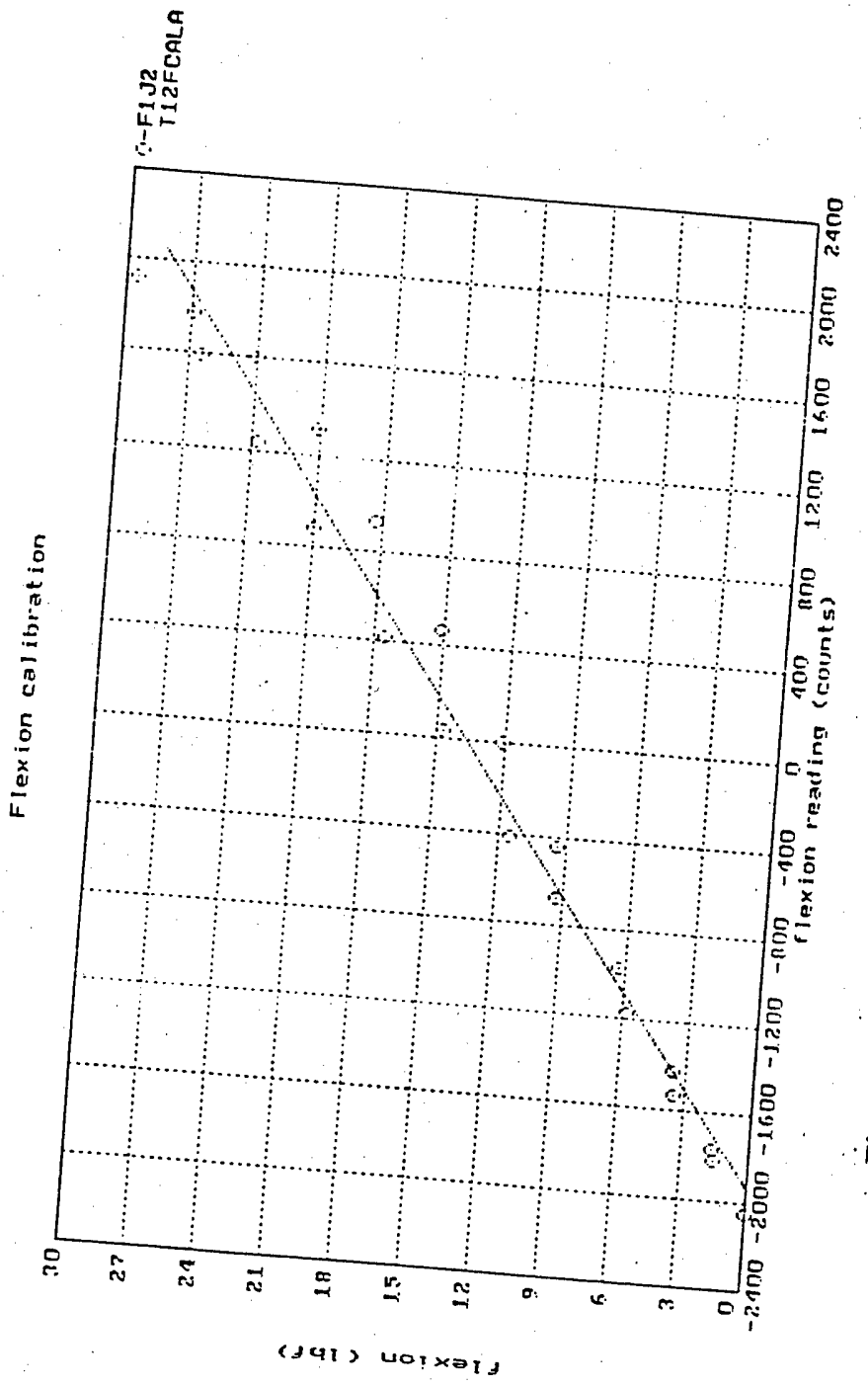



Figure B8 Flexion Tendon Sensor Calibration

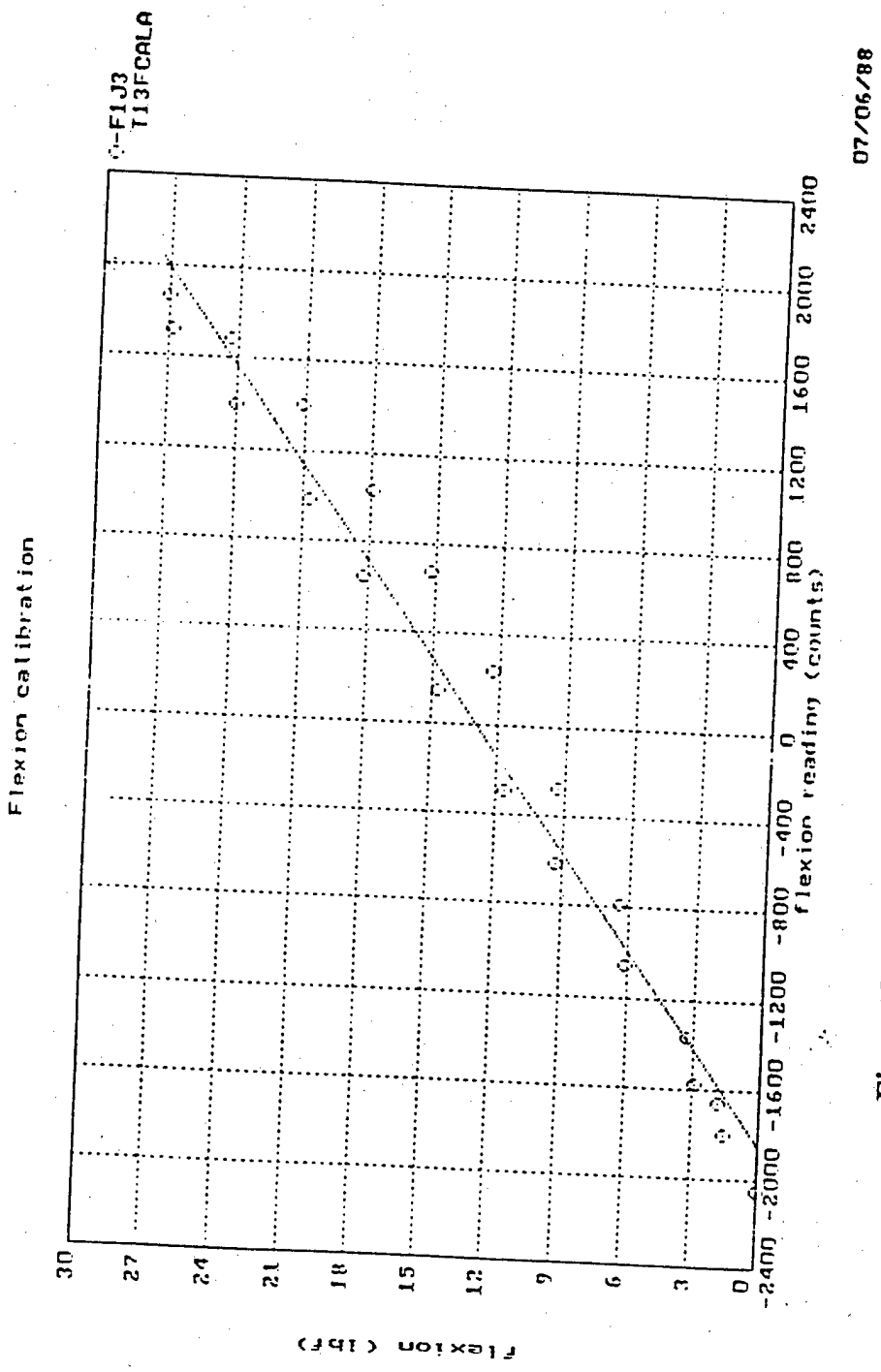



Figure B9 Flexion Tendon Sensor Calibration

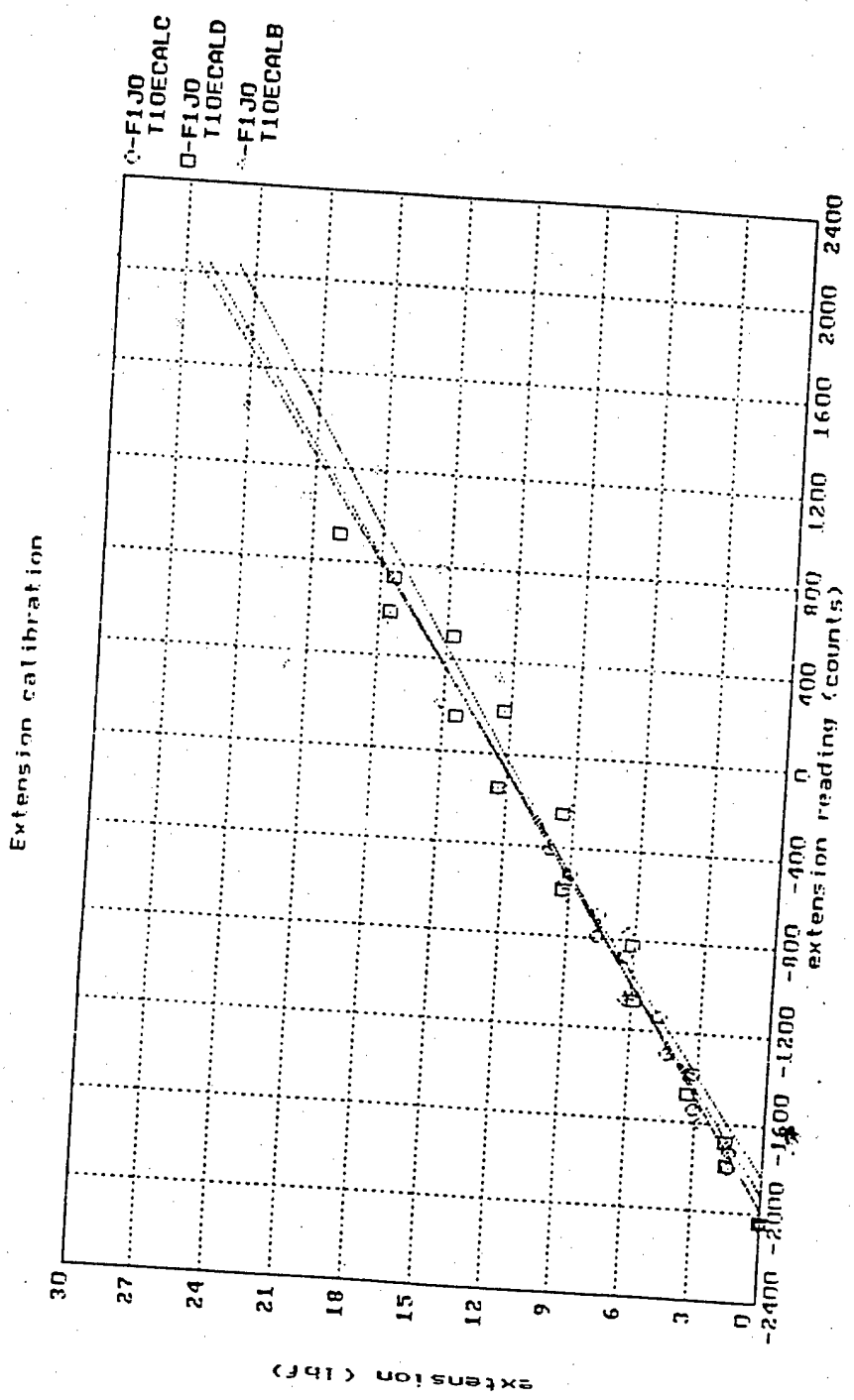



Figure B10 Extension Tendon Sensor Calibration

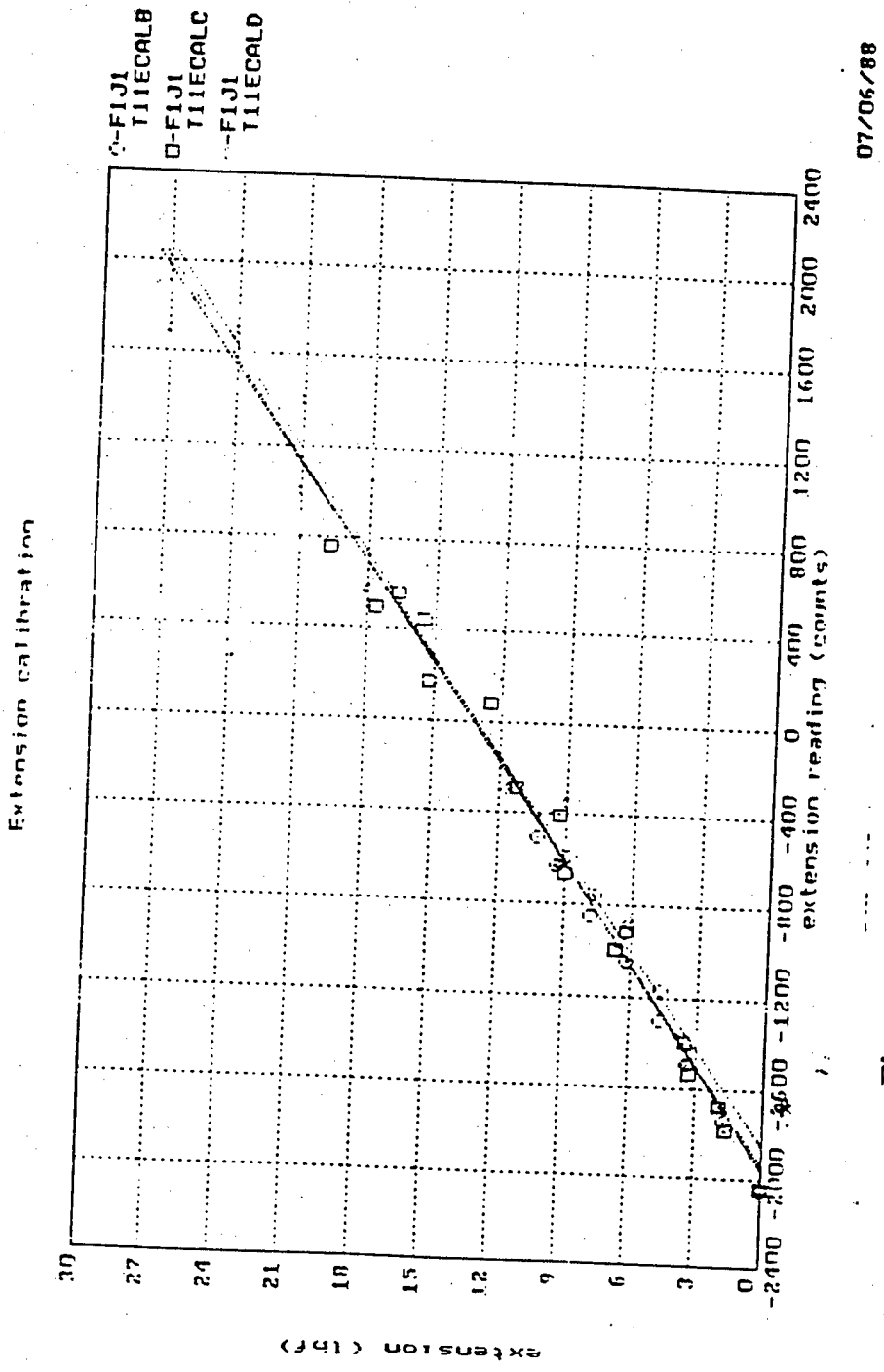



Figure B11 Extension Tendon Sensor Calibration

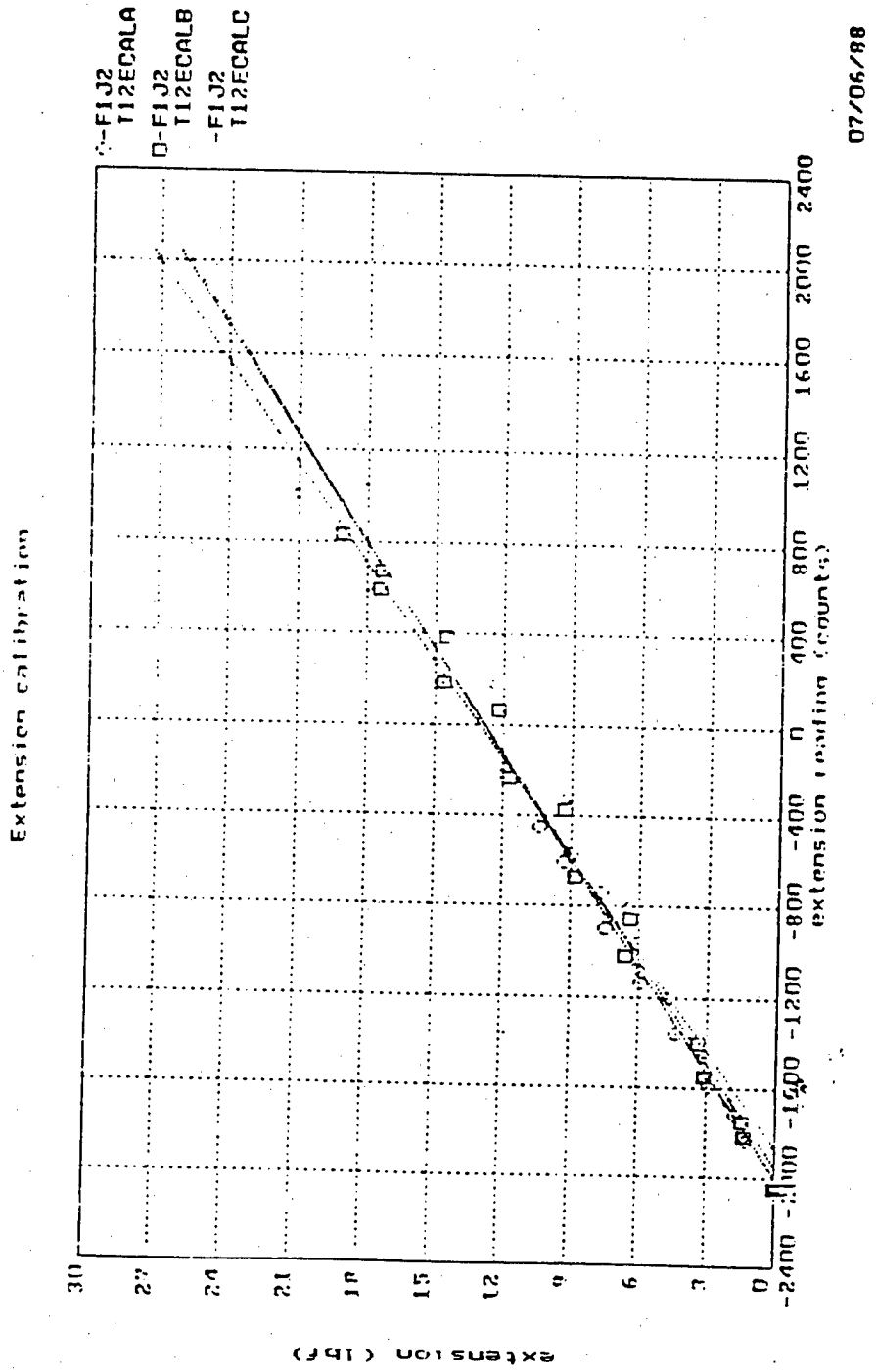



Figure B12 Extension Tendon Sensor Calibration

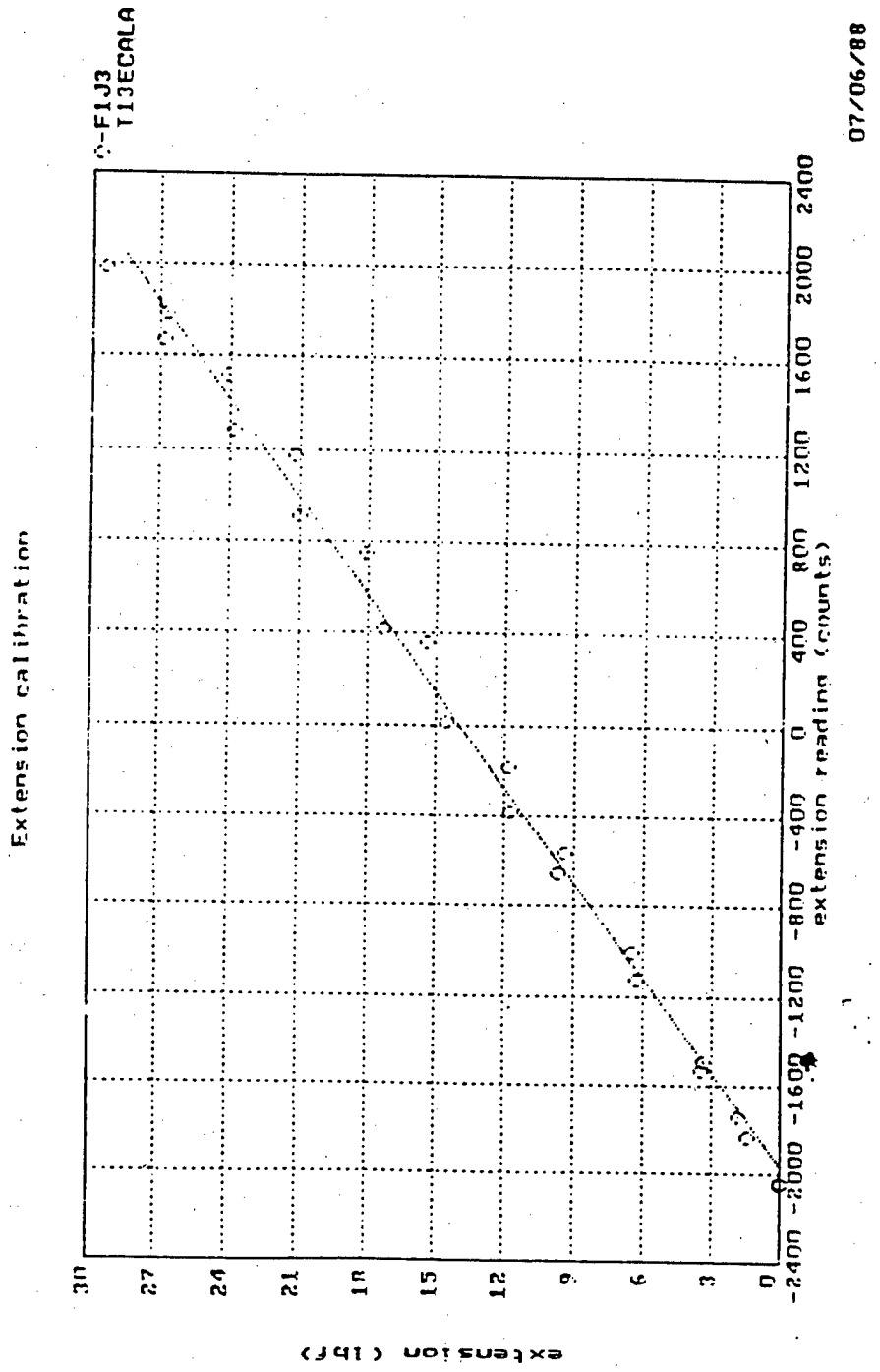



Figure B13 Extension Tendon Sensor Calibration

Raw File GP-VCAL01
Static calibration of postion & velocity gain control blocks
Finger 1, joint 0

| data set | desired input FD-F1J0 | actual input POS-F1J0 | input error (calc) | gained output (calc) | gain factor (calc) | gain voltage GP-F1J0 |
|---|---|---|---|---|---|---|
| 1 | 8.398 | 9.250 | -0.851 | -28.244 | 33.19 | 9.995 |
| 2 | 8.398 | 9.253 | -0.855 | -26.930 | 31.50 | 8.997 |
| 3 | 8.398 | 9.253 | -0.854 | -25.424 | 29.75 | 7.998 |
| 4 | 8.398 | 9.283 | -0.884 | -24.783 | 28.03 | 6.997 |
| 5 | 8.398 | 9.297 | -0.898 | -23.609 | 26.28 | 5.996 |
| 6 | 8.398 | 9.309 | -0.911 | -22.387 | 24.58 | 5.000 |
| 7 | 8.398 | 9.294 | -0.896 | -20.400 | 22.77 | 3.999 |
| 8 | 8.398 | 9.309 | -0.911 | -19.172 | 21.05 | 2.998 |
| 9 | 8.398 | 9.301 | -0.903 | -17.447 | 19.33 | 1.997 |
| 10 | 8.398 | 9.279 | -0.881 | -15.555 | 17.66 | 0.996 |
| 11 | 8.398 | 9.312 | -0.913 | -14.537 | 15.92 | 0.000 |
| 12 | 8.398 | 9.293 | -0.895 | -12.682 | 14.18 | -0.996 |
| 13 | 8.398 | 9.270 | -0.872 | -10.863 | 12.46 | -1.997 |
| 14 | 8.398 | 9.314 | -0.916 | -9.877 | 10.79 | -2.998 |
| 15 | 8.398 | 9.281 | -0.882 | -7.992 | 9.06 | -3.999 |
| 16 | 8.398 | 9.293 | -0.895 | -6.547 | 7.32 | -5.000 |
| 17 | 8.398 | 9.306 | -0.907 | -5.141 | 5.67 | -5.996 |
| 18 | 8.398 | 9.298 | -0.900 | -3.568 | 3.97 | -6.997 |
| 19 | 8.398 | 9.286 | -0.888 | -2.025 | 2.28 | -7.998 |
| 20 | 8.398 | 9.312 | -0.914 | -0.586 | 0.64 | -8.999 |
| 21 | 8.398 | 9.279 | -0.880 | -0.020 | 0.02 | -9.995 |
| 22 | -4.497 | -5.308 | 0.811 | 28.002 | 34.53 | 9.995 |
| 23 | -4.497 | -5.335 | 0.838 | 27.342 | 32.63 | 8.999 |
| 24 | -4.497 | -5.324 | 0.827 | 25.455 | 30.77 | 7.998 |
| 25 | -4.497 | -5.337 | 0.840 | 24.336 | 28.98 | 6.997 |
| 26 | -4.497 | -5.327 | 0.830 | 22.561 | 27.18 | 5.996 |
| 27 | -4.497 | -5.307 | 0.810 | 20.510 | 25.33 | 5.000 |
| 28 | -4.497 | -5.311 | 0.813 | 19.174 | 23.57 | 3.999 |
| 29 | -4.497 | -5.292 | 0.794 | 17.279 | 21.75 | 2.998 |
| 30 | -4.497 | -5.297 | 0.800 | 15.967 | 19.96 | 1.997 |
| 31 | -4.497 | -5.314 | 0.817 | 14.865 | 18.19 | 0.996 |
| 32 | -4.497 | -5.342 | 0.845 | 13.854 | 16.39 | 0.000 |
| 33 | -4.497 | -5.326 | 0.829 | 12.131 | 14.63 | -0.996 |
| 34 | -4.497 | -5.322 | 0.825 | 10.607 | 12.85 | -1.997 |
| 35 | -4.497 | -5.308 | 0.811 | 8.961 | 11.05 | -2.998 |
| 36 | -4.497 | -5.299 | 0.802 | 7.457 | 9.30 | -3.999 |
| 37 | -4.497 | -5.312 | 0.815 | 6.160 | 7.56 | -5.000 |
| 38 | -4.497 | -5.330 | 0.833 | 4.844 | 5.81 | -5.996 |
| 39 | -4.497 | -5.314 | 0.817 | 3.342 | 4.09 | -6.997 |
| 40 | -4.497 | -5.307 | 0.810 | 1.910 | 2.36 | -7.998 |
| 41 | -4.497 | -5.292 | 0.794 | 0.145 | 0.18 | -8.999 |
| 42 | -4.497 | -5.308 | 0.811 | -0.020 | -0.02 | -9.995 |

**Table B1** Calibration table of position gain block Finger 1 Joint 0

Raw File GPCAL.01
Static calibration of postion & velocity gain control blocks
Finger 1, joint 1

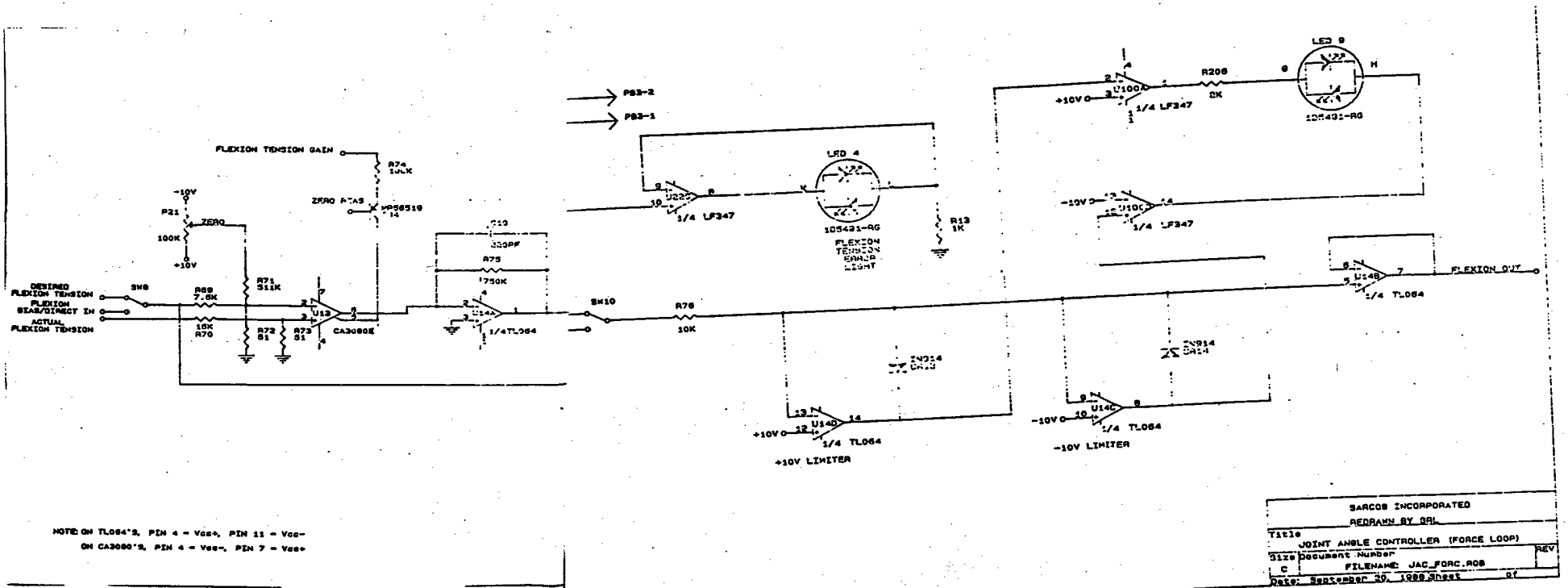
| data set | desired input PD-F1J1 | actual input POS-F1J1 | input error (calc) | gained output (calc) | gain factor (calc) | gain voltage GP-F1J1 |
|---|---|---|---|---|---|---|
| 1 | 7.598 | 8.457 | -0.859 | -30.926 | 36.01 | 9.995 |
| 2 | 7.598 | 8.457 | -0.859 | -29.426 | 34.24 | 8.999 |
| 3 | 7.598 | 8.454 | -0.856 | -27.668 | 32.32 | 7.998 |
| 4 | 7.598 | 8.478 | -0.880 | -26.834 | 30.50 | 6.997 |
| 5 | 7.598 | 8.461 | -0.864 | -24.775 | 28.68 | 5.996 |
| 6 | 7.598 | 8.469 | -0.872 | -23.383 | 26.83 | 5.000 |
| 7 | 7.598 | 8.463 | -0.865 | -21.678 | 25.05 | 3.999 |
| 8 | 7.598 | 8.482 | -0.884 | -20.506 | 23.19 | 2.998 |
| 9 | 7.598 | 8.468 | -0.870 | -18.576 | 21.35 | 1.997 |
| 10 | 7.598 | 8.467 | -0.869 | -16.957 | 19.51 | 0.996 |
| 11 | 7.598 | 8.478 | -0.880 | -15.590 | 17.71 | 0.000 |
| 12 | 7.598 | 8.469 | -0.872 | -13.846 | 15.87 | -0.996 |
| 13 | 7.598 | 8.466 | -0.869 | -12.223 | 14.07 | -1.997 |
| 14 | 7.598 | 8.483 | -0.885 | -10.812 | 12.22 | -2.998 |
| 15 | 7.598 | 8.463 | -0.865 | -9.035 | 10.45 | -3.999 |
| 16 | 7.598 | 8.479 | -0.882 | -7.582 | 8.60 | -5.000 |
| 17 | 7.598 | 8.460 | -0.862 | -5.803 | 6.82 | -5.996 |
| 18 | 7.598 | 8.470 | -0.873 | -4.350 | 4.98 | -6.997 |
| 19 | 7.598 | 8.455 | -0.857 | -2.748 | 3.21 | -7.998 |
| 20 | 7.598 | 8.466 | -0.868 | -1.223 | 1.41 | -8.999 |
| 21 | 7.598 | 8.451 | -0.853 | -0.020 | 0.02 | -9.995 |
| 22 | -4.800 | -5.670 | 0.871 | 31.861 | 36.60 | 9.995 |
| 23 | -4.800 | -5.677 | 0.877 | 30.346 | 34.60 | 8.999 |
| 24 | -4.800 | -5.677 | 0.877 | 28.740 | 32.75 | 7.998 |
| 25 | -4.800 | -5.676 | 0.876 | 27.014 | 30.82 | 6.997 |
| 26 | -4.800 | -5.675 | 0.875 | 25.363 | 28.97 | 5.996 |
| 27 | -4.800 | -5.665 | 0.865 | 23.445 | 27.11 | 5.000 |
| 28 | -4.800 | -5.663 | 0.863 | 21.789 | 25.24 | 3.999 |
| 29 | -4.800 | -5.661 | 0.861 | 20.127 | 23.37 | 2.998 |
| 30 | -4.800 | -5.660 | 0.860 | 18.455 | 21.46 | 1.997 |
| 31 | -4.800 | -5.675 | 0.875 | 17.141 | 19.59 | 0.996 |
| 32 | -4.800 | -5.689 | 0.890 | 15.744 | 17.70 | 0.000 |
| 33 | -4.800 | -5.678 | 0.878 | 13.951 | 15.88 | -0.996 |
| 34 | -4.800 | -5.681 | 0.881 | 12.371 | 14.04 | -1.997 |
| 35 | -4.800 | -5.670 | 0.871 | 10.637 | 12.22 | -2.998 |
| 36 | -4.800 | -5.667 | 0.867 | 8.998 | 10.38 | -3.999 |
| 37 | -4.800 | -5.665 | 0.865 | 7.404 | 8.56 | -5.000 |
| 38 | -4.800 | -5.679 | 0.879 | 5.936 | 6.75 | -5.996 |
| 39 | -4.800 | -5.673 | 0.873 | 4.314 | 4.94 | -6.997 |
| 40 | -4.800 | -5.676 | 0.876 | 2.760 | 3.15 | -7.998 |
| 41 | -4.800 | -5.670 | 0.870 | 1.197 | 1.38 | -8.999 |
| 42 | -4.800 | -5.672 | 0.873 | -0.020 | -0.02 | -9.995 |

Table B2 Calibration table of position gain block Finger 1 Joint 1

Raw File GP2CAL23
Static calibration of position & velocity gain control blocks
Finger 1, joint 2

| data set | desired input PD-FIJ2 | actual input POS-FIJ2 | input error (calc) | gained output (calc) | gain factor (calc) | gain voltage GP-FIJ2 |
|---|---|---|---|---|---|---|
| 1 | 0.096 | 0.978 | -0.882 | -31.707 | 35.94 | 9.995 |
| 2 | 0.096 | 0.981 | -0.885 | -30.078 | 33.98 | 8.999 |
| 3 | 0.096 | 0.963 | -0.868 | -27.965 | 32.23 | 7.998 |
| 4 | 0.096 | 0.965 | -0.869 | -26.387 | 30.36 | 6.997 |
| 5 | 0.096 | 0.958 | -0.862 | -24.516 | 28.45 | 5.996 |
| 6 | 0.096 | 0.982 | -0.887 | -23.520 | 26.52 | 5.000 |
| 7 | 0.096 | 0.983 | -0.887 | -21.883 | 24.66 | 3.999 |
| 8 | 0.096 | 0.982 | -0.887 | -20.225 | 22.81 | 2.998 |
| 9 | 0.096 | 0.988 | -0.893 | -18.770 | 21.03 | 1.997 |
| 10 | 0.096 | 0.976 | -0.880 | -16.852 | 19.15 | 0.996 |
| 11 | 0.096 | 0.972 | -0.876 | -15.217 | 17.37 | 0.000 |
| 12 | 0.096 | 0.959 | -0.863 | -13.461 | 15.59 | -0.996 |
| 13 | 0.096 | 0.969 | -0.874 | -12.021 | 13.76 | -1.997 |
| 14 | 0.096 | 0.965 | -0.869 | -10.379 | 11.94 | -2.998 |
| 15 | 0.096 | 0.966 | -0.871 | -8.816 | 10.13 | -3.999 |
| 16 | 0.096 | 0.967 | -0.874 | -7.258 | 8.31 | -5.000 |
| 17 | 0.096 | 0.985 | -0.889 | -5.797 | 6.52 | -5.996 |
| 18 | 0.096 | 0.975 | -0.879 | -4.164 | 4.74 | -6.997 |
| 19 | 0.096 | 0.969 | -0.874 | -2.621 | 3.00 | -7.998 |
| 20 | 0.096 | 0.974 | -0.878 | -1.107 | 1.26 | -8.999 |
| 21 | 0.096 | 0.970 | -0.874 | -0.020 | 0.02 | -9.995 |
| 22 | 0.996 | 0.128 | 0.868 | 31.141 | 35.87 | 9.995 |
| 23 | 0.996 | 0.113 | 0.883 | 30.002 | 33.97 | 8.999 |
| 24 | 0.996 | 0.109 | 0.887 | 28.404 | 32.03 | 7.998 |
| 25 | 0.996 | 0.117 | 0.879 | 26.578 | 30.22 | 6.997 |
| 26 | 0.996 | 0.110 | 0.886 | 25.166 | 28.41 | 5.996 |
| 27 | 0.996 | 0.117 | 0.879 | 23.369 | 26.59 | 5.000 |
| 28 | 0.996 | 0.122 | 0.874 | 21.629 | 24.75 | 3.999 |
| 29 | 0.996 | 0.115 | 0.881 | 20.174 | 22.89 | 2.998 |
| 30 | 0.996 | 0.112 | 0.884 | 18.652 | 21.09 | 1.997 |
| 31 | 0.996 | 0.114 | 0.882 | 17.010 | 19.20 | 0.996 |
| 32 | 0.996 | 0.118 | 0.878 | 15.350 | 17.47 | 0.000 |
| 33 | 0.996 | 0.099 | 0.897 | 14.006 | 15.61 | -0.996 |
| 34 | 0.996 | 0.108 | 0.888 | 12.297 | 13.85 | -1.997 |
| 35 | 0.996 | 0.109 | 0.887 | 10.684 | 12.04 | -2.998 |
| 36 | 0.996 | 0.111 | 0.885 | 9.094 | 10.27 | -3.999 |
| 37 | 0.996 | 0.110 | 0.886 | 7.471 | 8.43 | -5.000 |
| 38 | 0.996 | 0.115 | 0.881 | 5.873 | 6.67 | -5.996 |
| 39 | 0.996 | 0.120 | 0.876 | 4.254 | 4.86 | -6.997 |
| 40 | 0.996 | 0.118 | 0.878 | 2.697 | 3.07 | -7.998 |
| 41 | 0.996 | 0.133 | 0.863 | 1.127 | 1.31 | -8.999 |
| 42 | 0.996 | 0.121 | 0.875 | -0.020 | -0.02 | -9.995 |

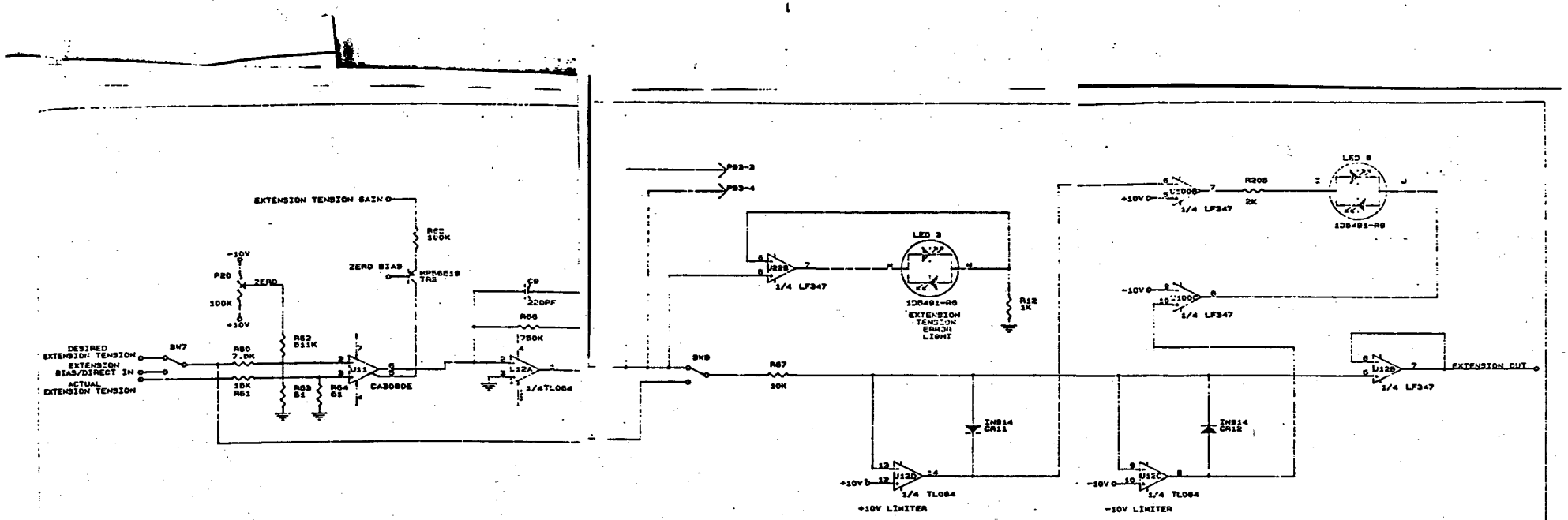
**Table B3** **Calibration table of position gain block Finger 1 Joint 2**

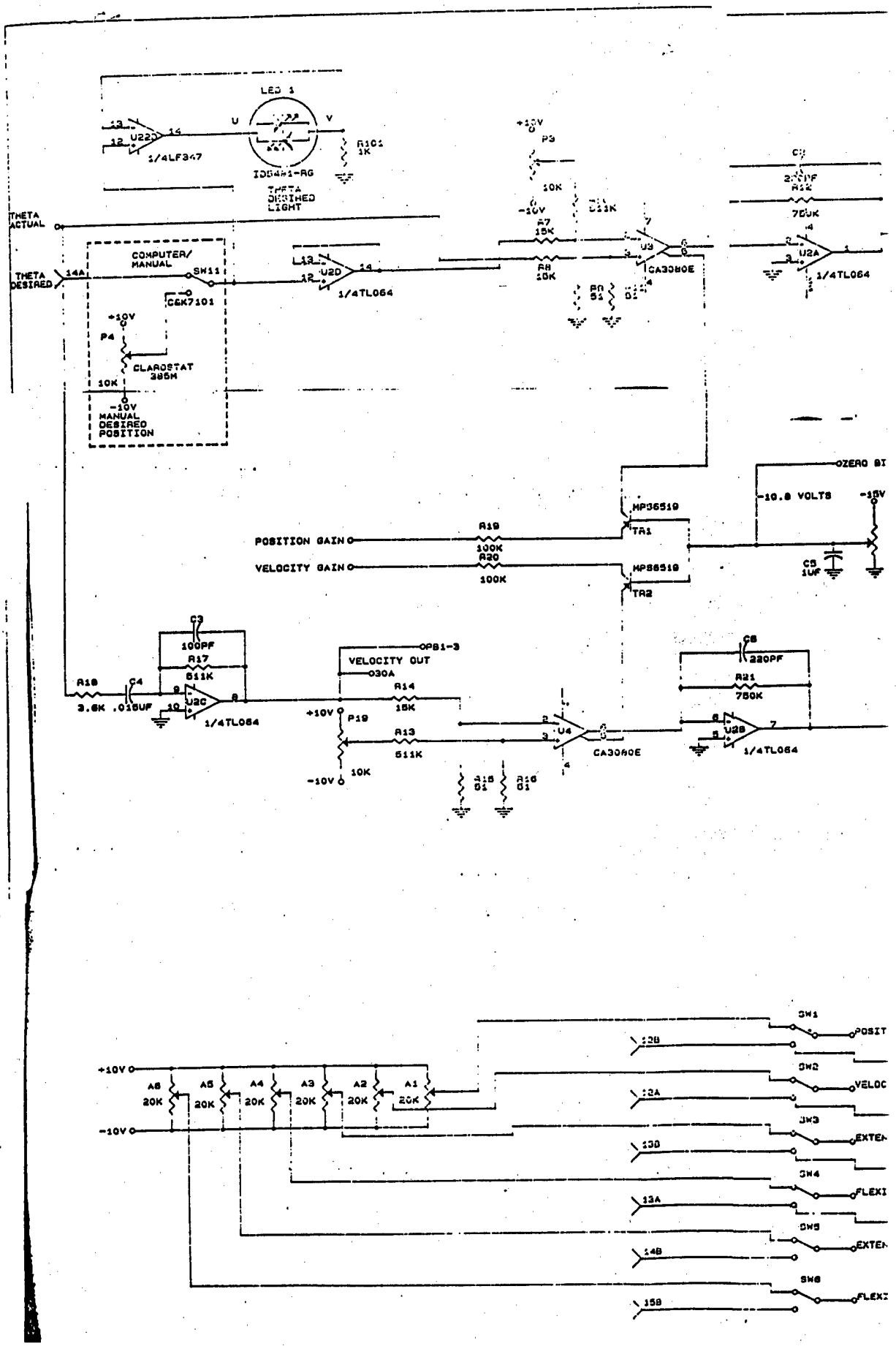



NOTE: ON TL064'S, PIN 4 = Vee+, PIN 11 = Vee-
ON CA3080'S, PIN 4 = Vee-, PIN 7 = Vee+

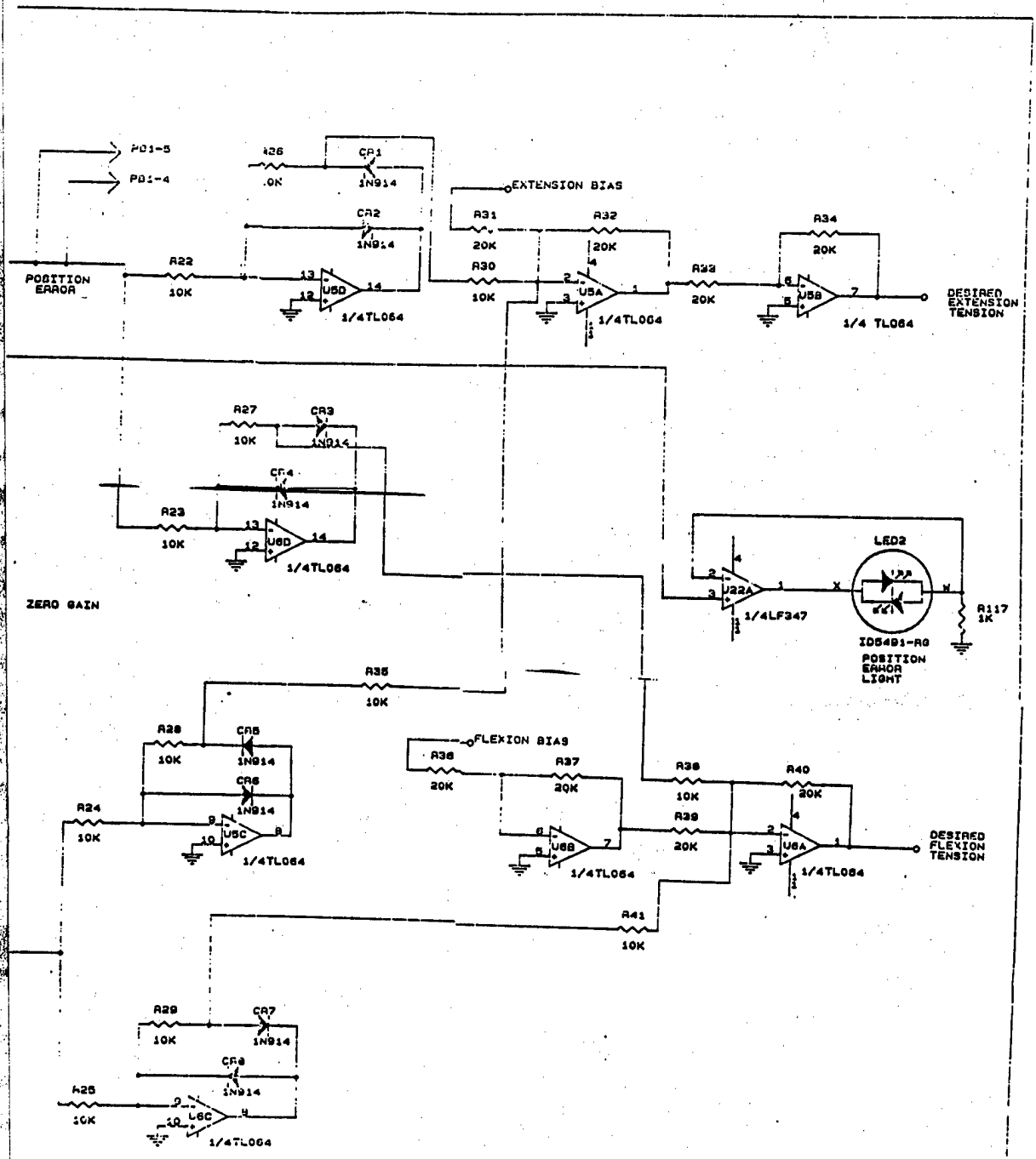
| | | |
|---|---|---|
| SARCOS INCORPORATED | | |
| REDRAWN BY DRL | | |
| Title: JOINT ANGLE CONTROLLER (FORCE LOOP) | | |
| Size: C | Document Number: FILENAME: JAC_FORC.ROB | REV |
| Date: September 29, 1988 | Sheet of | |

(1)


EXTENSION TENSION GAIN
R62
100K
-10V
P20
ZERO
100K
+10V
ZERO BIAS
MPS6519
TA3
C9
220PF
R66
750K
DESIRED
EXTENSION TENSION
EXTENSION
BIAS/DIRECT IN
ACTUAL
EXTENSION TENSION
SW7
R60
7.5K
R62
511K
15K
R61
R63
51
R64
51
U11
CA3080E
U12A
1/4TL064
P93-3
P93-4
U12B
1/4 LF347
LED 3
1D5491-R6
EXTENSION
TENSION
ENABLE
LIGHT
R12
1K
SW8
R67
10K
+10V LIMITER
-10V LIMITER
U12D
1/4 TL064
U12C
1/4 TL064
IN914
CR11
IN914
CR12
+10V
U12D
1/4 LF347
R205
2K
LED 8
1D5491-R6
-10V
U12C
1/4 LF347
EXTENSION OUT
U12B
1/4 LF347


LED 1
U
V
U22D
1/4LF347
R101
1K
THETA
DESIRED
LIGHT
THETA
ACTUAL
THETA
DESIRED
COMPUTER/
MANUAL
SW11
C6K7101
+10V
P4
CLAROSTAT
385M
10K
-10V
MANUAL
DESIRED
POSITION
U2D
1/4TL064
+10V
P3
10K
-10V
R7
15K
R8
10K
U3
CA3080E
R11
511K
R9
51
C1
2.2UF
R12
750K
U2A
1/4TL064
ZERO SI
-10.8 VOLTS
-15V
MPS6519
TR1
MPS6518
TR2
POSITION GAIN
VELOCITY GAIN
R19
100K
R20
100K
C5
1UF
C3
100PF
R17
511K
R18
3.6K
C4
.015UF
U2C
1/4TL064
OPB1-3
VELOCITY OUT
030A
R14
15K
+10V
P19
R13
511K
10K
-10V
U4
CA3080E
R15
51
R16
51
C8
220PF
R21
750K
U2B
1/4TL064
+10V
-10V
A6
20K
A5
20K
A4
20K
A3
20K
A2
20K
A1
20K
SW1
POSIT
13B
SW2
VELOC
12A
SW3
EXTEN
13B
SW4
FLEXI
13A
SW5
EXTEN
14B
SW6
FLEXI
15B

(3)




NOTE: ON TL084'S, PIN 4 = Vcc+, PIN 11 = Vcc-
ON CA3080'S, PIN 4 = Vee-, PIN 7 = Vee+

| | |
|---|---|
| SARCOS INCORPORATED | |
| REDRAWN BY SAL | |
| Title | JOINT ANGLE CONTROLLER (POS. VELOCITY LOOP) |
| Size C | Document Number FILENAME: JAC_POS.ROB REV |
| Date: September 19, 1990 | Sheet of |

Raw File GLFCAL01
Static calibration of extension & flexion gain control blocks
Finger 1, joint 0

| data set | desired input (calc) | actual input FLX-F1J0 | input error (calc) | gained output FG0-F1J0 | gain factor (calc) | gain voltage GF-F1J0 |
|---|---|---|---|---|---|---|
| 1 | -8.799 | -9.750 | 0.951 | 8.310 | 8.74 | 9.995 |
| 2 | -8.799 | -9.747 | 0.948 | 7.853 | 8.28 | 8.999 |
| 3 | -8.799 | -9.733 | 0.935 | 7.313 | 7.82 | 7.998 |
| 4 | -8.799 | -9.732 | 0.933 | 6.880 | 7.37 | 6.997 |
| 5 | -8.799 | -9.731 | 0.933 | 6.433 | 6.90 | 5.996 |
| 6 | -8.799 | -9.731 | 0.932 | 6.008 | 6.45 | 5.000 |
| 7 | -8.799 | -9.722 | 0.923 | 5.538 | 6.00 | 3.999 |
| 8 | -8.799 | -9.728 | 0.929 | 5.201 | 5.54 | 2.998 |
| 9 | -8.799 | -9.748 | 0.949 | 4.815 | 5.08 | 1.997 |
| 10 | -8.799 | -9.734 | 0.935 | 4.328 | 4.63 | 0.996 |
| 11 | -8.799 | -9.728 | 0.929 | 3.874 | 4.19 | 0.000 |
| 12 | -8.799 | -9.731 | 0.933 | 3.476 | 3.73 | -0.996 |
| 13 | -8.799 | -9.735 | 0.936 | 3.076 | 3.29 | -1.997 |
| 14 | -8.799 | -9.734 | 0.935 | 2.646 | 2.83 | -2.998 |
| 15 | -8.799 | -9.737 | 0.938 | 2.232 | 2.38 | -3.999 |
| 16 | -8.799 | -9.721 | 0.922 | 1.786 | 1.94 | -5.000 |
| 17 | -8.799 | -9.729 | 0.930 | 1.386 | 1.49 | -5.996 |
| 18 | -8.799 | -9.727 | 0.928 | 0.971 | 1.05 | -6.997 |
| 19 | -8.799 | -9.739 | 0.940 | 0.566 | 0.60 | -7.998 |
| 20 | -8.799 | -9.730 | 0.932 | 0.158 | 0.17 | -8.999 |
| 21 | -8.799 | -9.732 | 0.933 | 0.000 | 0.00 | -9.995 |

Table B5 Calibration table of flexion gain block
Finger 1 Joint 0

Raw File GEFCAL01
Static calibration of extension & flexion gain control blocks
Finger 1, joint 1

| data set | desired input (calc) | actual input FLX-F1J1 | input error (calc) | gained output FGO-F1J1 | gain factor (calc) | gain voltage GF-F1J1 |
|---|---|---|---|---|---|---|
| 1 | -8.496 | -9.503 | 1.007 | 7.942 | 7.89 | 9.995 |
| 2 | -8.496 | -9.491 | 0.995 | 7.431 | 7.47 | 8.999 |
| 3 | -8.496 | -9.505 | 1.009 | 7.158 | 7.10 | 7.998 |
| 4 | -8.496 | -9.501 | 1.005 | 6.701 | 6.67 | 6.997 |
| 5 | -8.496 | -9.497 | 1.001 | 6.276 | 6.27 | 5.996 |
| 6 | -8.496 | -9.496 | 1.000 | 5.854 | 5.86 | 5.000 |
| 7 | -8.496 | -9.481 | 0.985 | 5.370 | 5.45 | 3.999 |
| 8 | -8.496 | -9.506 | 1.010 | 5.121 | 5.07 | 2.998 |
| 9 | -8.496 | -9.503 | 1.007 | 4.698 | 4.66 | 1.997 |
| 10 | -8.496 | -9.493 | 0.997 | 4.243 | 4.26 | 0.996 |
| 11 | -8.496 | -9.487 | 0.991 | 3.824 | 3.86 | 0.000 |
| 12 | -8.496 | -9.496 | 1.000 | 3.461 | 3.46 | -0.996 |
| 13 | -8.496 | -9.490 | 0.994 | 3.048 | 3.07 | -1.997 |
| 14 | -8.496 | -9.496 | 1.000 | 2.670 | 2.67 | -2.998 |
| 15 | -8.496 | -9.489 | 0.993 | 2.259 | 2.27 | -3.999 |
| 16 | -8.496 | -9.478 | 0.982 | 1.837 | 1.87 | -5.000 |
| 17 | -8.496 | -9.491 | 0.995 | 1.476 | 1.48 | -5.996 |
| 18 | -8.496 | -9.491 | 0.995 | 1.083 | 1.09 | -6.997 |
| 19 | -8.496 | -9.493 | 0.997 | 0.698 | 0.70 | -7.998 |
| 20 | -8.496 | -9.489 | 0.993 | 0.310 | 0.31 | -8.999 |
| 21 | -8.496 | -9.476 | 0.979 | -0.005 | -0.00 | -9.995 |

Table B6 Calibration table of flexion gain block Finger 1 Joint 1

Raw File GEFCAL23
Static calibration of extension & flexion gain control blocks
Finger 1, joint 2

| data set | desired input (calc) | actual input FLX-F1J2 | input error (calc) | gained output FGO-F1J2 | gain factor (calc) | gain voltage GF-F1J2 |
|---|---|---|---|---|---|---|
| 1 | -8.398 | -9.290 | 0.891 | 8.067 | 9.05 | 9.995 |
| 2 | -8.398 | -9.292 | 0.894 | 7.660 | 8.57 | 8.999 |
| 3 | -8.398 | -9.293 | 0.895 | 7.260 | 8.12 | 7.990 |
| 4 | -8.398 | -9.292 | 0.894 | 6.838 | 7.65 | 6.997 |
| 5 | -8.398 | -9.292 | 0.893 | 6.405 | 7.17 | 5.996 |
| 6 | -8.398 | -9.285 | 0.886 | 5.948 | 6.71 | 5.000 |
| 7 | -8.398 | -9.290 | 0.891 | 5.577 | 6.26 | 3.999 |
| 8 | -8.398 | -9.287 | 0.889 | 5.128 | 5.77 | 2.998 |
| 9 | -8.398 | -9.287 | 0.889 | 4.731 | 5.32 | 1.997 |
| 10 | -8.398 | -9.299 | 0.901 | 4.367 | 4.85 | 0.996 |
| 11 | -8.398 | -9.294 | 0.896 | 3.947 | 4.41 | 0.000 |
| 12 | -8.398 | -9.292 | 0.894 | 3.520 | 3.94 | -0.996 |
| 13 | -8.398 | -9.289 | 0.890 | 3.099 | 3.48 | -1.997 |
| 14 | -8.398 | -9.290 | 0.892 | 2.694 | 3.02 | -2.998 |
| 15 | -8.398 | -9.293 | 0.895 | 2.296 | 2.57 | -3.999 |
| 16 | -8.398 | -9.292 | 0.894 | 1.889 | 2.11 | -5.000 |
| 17 | -8.398 | -9.292 | 0.894 | 1.486 | 1.66 | -5.996 |
| 18 | -8.398 | -9.289 | 0.890 | 1.078 | 1.21 | -6.997 |
| 19 | -8.398 | -9.286 | 0.888 | 0.678 | 0.76 | -7.998 |
| 20 | -8.398 | -9.280 | 0.881 | 0.282 | 0.32 | -8.999 |
| 21 | -8.398 | -9.283 | 0.885 | -0.007 | -0.00 | -9.995 |

**Table B7** Calibration table of flexion gain block Finger 1 Joint 2

Raw File GEFCAL23
Static calibration of extension & flexion gain control blocks
Finger 1, joint 3

| data set | desired input (calc) | actual input FLX-FIJ3 | input error (calc) | gained output FGO-FIJ3 | gain factor (calc) | gain voltage GF-FIJ3 |
|---|---|---|---|---|---|---|
| 1 | -9.092 | -10.000 | 0.908 | 8.992 | 9.90 | 9.995 |
| 2 | -9.092 | -10.000 | 0.908 | 8.483 | 9.34 | 8.999 |
| 3 | -9.092 | -10.000 | 0.908 | 8.002 | 8.81 | 7.998 |
| 4 | -9.092 | -10.000 | 0.908 | 7.599 | 8.37 | 6.997 |
| 5 | -9.092 | -10.000 | 0.908 | 7.136 | 7.86 | 5.996 |
| 6 | -9.092 | -10.000 | 0.908 | 6.663 | 7.34 | 5.000 |
| 7 | -9.092 | -10.000 | 0.908 | 6.205 | 6.83 | 3.999 |
| 8 | -9.092 | -10.000 | 0.908 | 5.749 | 6.33 | 2.998 |
| 9 | -9.092 | -10.000 | 0.908 | 5.236 | 5.77 | 1.997 |
| 10 | -9.092 | -10.000 | 0.908 | 4.829 | 5.32 | 0.996 |
| 11 | -9.092 | -10.000 | 0.908 | 4.349 | 4.79 | 0.000 |
| 12 | -9.092 | -10.000 | 0.908 | 3.928 | 4.32 | -0.996 |
| 13 | -9.092 | -10.000 | 0.908 | 3.466 | 3.82 | -1.997 |
| 14 | -9.092 | -10.000 | 0.908 | 2.988 | 3.29 | -2.998 |
| 15 | -9.092 | -10.000 | 0.908 | 2.554 | 2.81 | -3.999 |
| 16 | -9.092 | -10.000 | 0.908 | 2.126 | 2.34 | -5.000 |
| 17 | -9.092 | -10.000 | 0.908 | 1.676 | 1.85 | -5.996 |
| 18 | -9.092 | -10.000 | 0.908 | 1.229 | 1.35 | -6.997 |
| 19 | -9.092 | -10.000 | 0.908 | 0.785 | 0.86 | -7.998 |
| 20 | -9.092 | -10.000 | 0.908 | 0.340 | 0.37 | -8.999 |
| 21 | -9.092 | -10.000 | 0.908 | -0.005 | -0.00 | -9.995 |

Table B8 Calibration table of flexion gain block Finger 1 Joint 3

Raw File DEFCAL01
Static calibration of extension & flexion gain control blocks
Finger 1, joint 0

| data set | desired input (calc) | actual input FLX-FIJ3 | input error (calc) | gained output FGO-FIJ3 | gain factor (calc) | gain voltage GF-FIJ3 |
|---|---|---|---|---|---|---|
| 1 | -9.092 | -10.000 | 0.908 | 8.992 | 9.90 | 9.995 |
| 2 | -9.092 | -10.000 | 0.908 | 8.483 | 9.34 | 8.999 |
| 3 | -9.092 | -10.000 | 0.908 | 8.002 | 8.81 | 7.998 |
| 4 | -9.092 | -10.000 | 0.908 | 7.599 | 8.37 | 6.997 |
| 5 | -9.092 | -10.000 | 0.908 | 7.136 | 7.86 | 5.996 |
| 6 | -9.092 | -10.000 | 0.908 | 6.663 | 7.34 | 5.000 |
| 7 | -9.092 | -10.000 | 0.908 | 6.205 | 6.83 | 3.999 |
| 8 | -9.092 | -10.000 | 0.908 | 5.749 | 6.33 | 2.998 |
| 9 | -9.092 | -10.000 | 0.908 | 5.236 | 5.77 | 1.997 |
| 10 | -9.092 | -10.000 | 0.908 | 4.829 | 5.32 | 0.996 |
| 11 | -9.092 | -10.000 | 0.908 | 4.349 | 4.79 | 0.000 |
| 12 | -9.092 | -10.000 | 0.908 | 3.928 | 4.32 | -0.996 |
| 13 | -9.092 | -10.000 | 0.908 | 3.466 | 3.82 | -1.997 |
| 14 | -9.092 | -10.000 | 0.908 | 2.988 | 3.29 | -2.998 |
| 15 | -9.092 | -10.000 | 0.908 | 2.554 | 2.81 | -3.999 |
| 16 | -9.092 | -10.000 | 0.908 | 2.126 | 2.34 | -5.000 |
| 17 | -9.092 | -10.000 | 0.908 | 1.676 | 1.85 | -5.996 |
| 18 | -9.092 | -10.000 | 0.908 | 1.229 | 1.35 | -6.997 |
| 19 | -9.092 | -10.000 | 0.908 | 0.785 | 0.86 | -7.998 |
| 20 | -9.092 | -10.000 | 0.908 | 0.340 | 0.37 | -8.999 |
| 21 | -9.092 | -10.000 | 0.908 | -0.005 | -0.00 | -9.995 |

Table B9 Calibration table of extension gain block Finger 1 Joint 0

Raw File GEFCAL01
Static calibration of extension & flexion gain control blocks
Finger 1, joint 1

| data set | desired input (calc) | actual input EXT-F1J1 | input error (calc) | gained output EGO-F1J1 | gain factor (calc) | gain voltage GE-F1J1 |
|---|---|---|---|---|---|---|
| 1 | -8.799 | -9.814 | 1.015 | 8.709 | 8.58 | 9.995 |
| 2 | -8.799 | -9.797 | 0.999 | 8.110 | 8.12 | 8.999 |
| 3 | -8.799 | -9.807 | 1.008 | 7.728 | 7.67 | 7.998 |
| 4 | -8.799 | -9.805 | 1.006 | 7.254 | 7.21 | 6.997 |
| 5 | -8.799 | -9.802 | 1.003 | 6.797 | 6.78 | 5.996 |
| 6 | -8.799 | -9.805 | 1.006 | 6.366 | 6.33 | 5.000 |
| 7 | -8.799 | -9.797 | 0.998 | 5.882 | 5.89 | 3.999 |
| 8 | -8.799 | -9.816 | 1.018 | 5.546 | 5.45 | 2.998 |
| 9 | -8.799 | -9.811 | 1.012 | 5.075 | 5.02 | 1.997 |
| 10 | -8.799 | -9.812 | 1.013 | 4.643 | 4.58 | 0.996 |
| 11 | -8.799 | -9.809 | 1.010 | 4.180 | 4.14 | 0.000 |
| 12 | -8.799 | -9.813 | 1.014 | 3.766 | 3.71 | -0.996 |
| 13 | -8.799 | -9.804 | 1.005 | 3.298 | 3.28 | -1.997 |
| 14 | -8.799 | -9.812 | 1.013 | 2.888 | 2.85 | -2.998 |
| 15 | -8.799 | -9.800 | 1.001 | 2.421 | 2.42 | -3.999 |
| 16 | -8.799 | -9.791 | 0.992 | 1.973 | 1.99 | -5.000 |
| 17 | -8.799 | -9.810 | 1.011 | 1.589 | 1.57 | -5.996 |
| 18 | -8.799 | -9.808 | 1.009 | 1.163 | 1.15 | -6.997 |
| 19 | -8.799 | -9.809 | 1.010 | 0.743 | 0.74 | -7.998 |
| 20 | -8.799 | -9.816 | 1.018 | 0.328 | 0.32 | -8.999 |
| 21 | -8.799 | -9.808 | 1.009 | -0.005 | -0.00 | -9.995 |

Table B10 Calibration table of extension gain block Finger 1 Joint 1

Raw File GEFCAL23
Static calibration of extension & flexion gain control blocks
Finger 1, joint 2

| data set | desired input (calc) | actual input EXT-FIJ2 | input error (calc) | gained output EGO-FIJ2 | gain factor (calc) | gain voltage GE-FIJ2 |
|---|---|---|---|---|---|---|
| 1 | -9.092 | -10.000 | 0.908 | 9.995 | 11.01 | 9.995 |
| 2 | -9.092 | -10.000 | 0.908 | 9.585 | 10.55 | 8.999 |
| 3 | -9.092 | -10.000 | 0.908 | 9.043 | 9.96 | 7.998 |
| 4 | -9.092 | -10.000 | 0.908 | 8.579 | 9.45 | 6.997 |
| 5 | -9.092 | -10.000 | 0.908 | 7.985 | 8.79 | 5.996 |
| 6 | -9.092 | -10.000 | 0.908 | 7.441 | 8.19 | 5.000 |
| 7 | -9.092 | -10.000 | 0.908 | 6.965 | 7.67 | 3.999 |
| 8 | -9.092 | -10.000 | 0.908 | 6.420 | 7.07 | 2.998 |
| 9 | -9.092 | -10.000 | 0.908 | 5.877 | 6.47 | 1.997 |
| 10 | -9.092 | -10.000 | 0.908 | 5.412 | 5.96 | 0.996 |
| 11 | -9.092 | -10.000 | 0.908 | 4.908 | 5.40 | 0.000 |
| 12 | -9.092 | -10.000 | 0.908 | 4.404 | 4.85 | -0.996 |
| 13 | -9.092 | -10.000 | 0.908 | 3.899 | 4.29 | -1.997 |
| 14 | -9.092 | -10.000 | 0.908 | 3.377 | 3.72 | -2.998 |
| 15 | -9.092 | -10.000 | 0.908 | 2.879 | 3.17 | -3.999 |
| 16 | -9.092 | -10.000 | 0.908 | 2.366 | 2.61 | -5.000 |
| 17 | -9.092 | -10.000 | 0.908 | 1.862 | 2.05 | -5.996 |
| 18 | -9.092 | -10.000 | 0.908 | 1.366 | 1.50 | -6.997 |
| 19 | -9.092 | -10.000 | 0.908 | 0.857 | 0.94 | -7.998 |
| 20 | -9.092 | -10.000 | 0.908 | 0.359 | 0.40 | -8.999 |
| 21 | -9.092 | -10.000 | 0.908 | -0.005 | -0.00 | -9.995 |

Table B11 Calibration table of extension gain block
Finger 1 Joint 2

Raw File GEFCAL23
Static calibration of extension & flexion gain control blocks
Finger 1, joint 3

| data set | desired input (calc) | actual input EXT-FIJ3 | input error (calc) | gained output EGU-FIJ3 | gain factor (calc) | gain voltage GE-FIJ3 |
|---|---|---|---|---|---|---|
| 1 | -9.092 | -10.000 | 0.908 | 7.622 | 8.39 | 9.995 |
| 2 | -9.092 | -10.000 | 0.908 | 7.145 | 7.87 | 8.999 |
| 3 | -9.092 | -10.000 | 0.908 | 6.707 | 7.47 | 7.998 |
| 4 | -9.092 | -10.000 | 0.908 | 6.435 | 7.09 | 6.997 |
| 5 | -9.092 | -10.000 | 0.908 | 6.057 | 6.67 | 5.996 |
| 6 | -9.092 | -10.000 | 0.908 | 5.644 | 6.21 | 5.000 |
| 7 | -9.092 | -10.000 | 0.908 | 5.279 | 5.81 | 3.999 |
| 8 | -9.092 | -10.000 | 0.908 | 4.885 | 5.38 | 2.998 |
| 9 | -9.092 | -10.000 | 0.908 | 4.450 | 4.90 | 1.997 |
| 10 | -9.092 | -10.000 | 0.908 | 4.093 | 4.51 | 0.996 |
| 11 | -9.092 | -10.000 | 0.908 | 3.708 | 4.08 | 0.000 |
| 12 | -9.092 | -10.000 | 0.908 | 3.354 | 3.69 | -0.996 |
| 13 | -9.092 | -10.000 | 0.908 | 2.945 | 3.24 | -1.997 |
| 14 | -9.092 | -10.000 | 0.908 | 2.578 | 2.84 | -2.998 |
| 15 | -9.092 | -10.000 | 0.908 | 2.185 | 2.41 | -3.999 |
| 16 | -9.092 | -10.000 | 0.908 | 1.798 | 1.98 | -5.000 |
| 17 | -9.092 | -10.000 | 0.900 | 1.435 | 1.58 | -5.996 |
| 18 | -9.092 | -10.000 | 0.908 | 1.050 | 1.16 | -6.997 |
| 19 | -9.092 | -10.000 | 0.908 | 0.673 | 0.74 | -7.998 |
| 20 | -9.092 | -10.000 | 0.908 | 0.276 | 0.33 | -8.999 |
| 21 | -9.092 | -10.000 | 0.908 | -0.003 | -0.00 | -9.995 |

**Table B12 Calibration table of extension gain block Finger 1 Joint 3**

Raw File GAVGAL01
Static calibration of postion & velocity gain control blocks
Finger 1, joint 0

| data set | desired input VD-FIJ0 | actual input VEL-FIJ0 | input error (calc) | gained output (calc) | gain factor (calc) | gain voltage GV-FIJ0 |
|---|---|---|---|---|---|---|
| 1 | 0.898 | 0.897 | 0.897 | 31.070 | 34.62 | 9.995 |
| 2 | 0.898 | 0.897 | 0.897 | 29.441 | 32.82 | 8.999 |
| 3 | 0.898 | 0.898 | 0.898 | 27.812 | 30.97 | 7.998 |
| 4 | 0.898 | 0.896 | 0.896 | 26.178 | 29.23 | 6.997 |
| 5 | 0.898 | 0.897 | 0.897 | 24.557 | 27.36 | 5.996 |
| 6 | 0.898 | 0.897 | 0.897 | 22.945 | 25.57 | 5.000 |
| 7 | 0.898 | 0.896 | 0.896 | 21.328 | 23.82 | 3.997 |
| 8 | 0.898 | 0.897 | 0.897 | 19.713 | 21.98 | 2.998 |
| 9 | 0.898 | 0.898 | 0.898 | 18.105 | 20.15 | 1.997 |
| 10 | 0.898 | 0.898 | 0.898 | 16.490 | 18.36 | 0.996 |
| 11 | 0.898 | 0.897 | 0.897 | 14.912 | 16.62 | 0.000 |
| 12 | 0.898 | 0.897 | 0.897 | 13.311 | 14.84 | -0.996 |
| 13 | 0.898 | 0.898 | 0.898 | 11.699 | 13.02 | -1.997 |
| 14 | 0.898 | 0.898 | 0.898 | 10.107 | 11.25 | -2.998 |
| 15 | 0.898 | 0.898 | 0.898 | 8.516 | 9.48 | -3.999 |
| 16 | 0.898 | 0.896 | 0.896 | 6.914 | 7.71 | -5.000 |
| 17 | 0.898 | 0.897 | 0.897 | 5.340 | 5.95 | -5.996 |
| 18 | 0.898 | 0.898 | 0.898 | 3.744 | 4.17 | -6.997 |
| 19 | 0.898 | 0.898 | 0.898 | 2.176 | 2.42 | -7.998 |
| 20 | 0.898 | 0.898 | 0.898 | 0.619 | 0.69 | -8.999 |
| 21 | 0.898 | 0.897 | 0.897 | 0.000 | 0.00 | -9.995 |
| 22 | -0.898 | -0.894 | -0.894 | -30.291 | 33.88 | 9.995 |
| 23 | -0.898 | -0.894 | -0.894 | -28.650 | 32.06 | 8.999 |
| 24 | -0.898 | -0.896 | -0.896 | -27.027 | 30.18 | 7.998 |
| 25 | -0.898 | -0.894 | -0.894 | -25.414 | 28.43 | 6.997 |
| 26 | -0.898 | -0.896 | -0.896 | -23.809 | 26.59 | 5.996 |
| 27 | -0.898 | -0.895 | -0.895 | -22.227 | 24.85 | 5.000 |
| 28 | -0.898 | -0.895 | -0.895 | -20.625 | 23.04 | 3.999 |
| 29 | -0.898 | -0.896 | -0.896 | -19.043 | 21.26 | 2.998 |
| 30 | -0.898 | -0.894 | -0.894 | -17.467 | 19.54 | 1.997 |
| 31 | -0.898 | -0.895 | -0.895 | -15.898 | 17.76 | 0.996 |
| 32 | -0.898 | -0.896 | -0.896 | -14.330 | 16.00 | 0.000 |
| 33 | -0.898 | -0.895 | -0.895 | -12.775 | 14.27 | -0.996 |
| 34 | -0.898 | -0.896 | -0.896 | -11.215 | 12.52 | -1.997 |
| 35 | -0.898 | -0.895 | -0.895 | -9.668 | 10.81 | -2.998 |
| 36 | -0.898 | -0.894 | -0.894 | -8.119 | 9.09 | -3.999 |
| 37 | -0.898 | -0.896 | -0.896 | -6.582 | 7.35 | -5.000 |
| 38 | -0.898 | -0.894 | -0.894 | -5.047 | 5.65 | -5.996 |
| 39 | -0.898 | -0.894 | -0.894 | -3.529 | 3.95 | -6.997 |
| 40 | -0.898 | -0.895 | -0.895 | -2.031 | 2.27 | -7.998 |
| 41 | -0.898 | -0.894 | -0.894 | -0.559 | 0.62 | -8.997 |
| 42 | -0.898 | -0.894 | -0.894 | 0.002 | -0.00 | -9.995 |

**Table B13** Calibration table of velocity gain block Finger 1 Joint 0

Raw File GAVCAL01
Static calibration of postion & velocity gain control blocks
Finger 1, joint 1

| data set | desired input VD-FIJ1 | actual input VEL-FIJ1 | input error (calc) | gained output (calc) | gain factor (calc) | gain voltage GV-FIJ1 |
|---|---|---|---|---|---|---|
| 1 | 0.898 | 0.896 | 0.896 | 30.586 | 34.12 | 9.995 |
| 2 | 0.898 | 0.897 | 0.897 | 29.021 | 32.34 | 8.999 |
| 3 | 0.898 | 0.897 | 0.897 | 27.469 | 30.61 | 7.998 |
| 4 | 0.898 | 0.898 | 0.898 | 25.938 | 28.89 | 6.997 |
| 5 | 0.898 | 0.897 | 0.897 | 24.375 | 27.16 | 5.996 |
| 6 | 0.898 | 0.896 | 0.896 | 22.840 | 25.49 | 5.000 |
| 7 | 0.898 | 0.897 | 0.897 | 21.287 | 23.72 | 3.999 |
| 8 | 0.898 | 0.897 | 0.897 | 19.736 | 21.99 | 2.998 |
| 9 | 0.898 | 0.897 | 0.897 | 18.193 | 20.28 | 1.997 |
| 10 | 0.898 | 0.896 | 0.896 | 16.641 | 18.57 | 0.996 |
| 11 | 0.898 | 0.898 | 0.898 | 15.111 | 16.83 | 0.000 |
| 12 | 0.898 | 0.897 | 0.897 | 13.559 | 15.12 | -0.996 |
| 13 | 0.898 | 0.896 | 0.896 | 12.012 | 13.41 | -1.997 |
| 14 | 0.898 | 0.897 | 0.897 | 10.465 | 11.66 | -2.998 |
| 15 | 0.898 | 0.897 | 0.897 | 8.908 | 9.93 | -3.999 |
| 16 | 0.898 | 0.898 | 0.898 | 7.375 | 8.21 | -5.000 |
| 17 | 0.898 | 0.898 | 0.898 | 5.824 | 6.49 | -5.996 |
| 18 | 0.898 | 0.898 | 0.898 | 4.279 | 4.77 | -6.997 |
| 19 | 0.898 | 0.898 | 0.898 | 2.732 | 3.04 | -7.998 |
| 20 | 0.898 | 0.897 | 0.897 | 1.199 | 1.34 | -8.999 |
| 21 | 0.898 | 0.897 | 0.897 | 0.000 | 0.00 | -9.995 |
| 22 | -0.898 | -0.898 | -0.898 | -30.762 | 34.26 | 9.995 |
| 23 | -0.898 | -0.896 | -0.896 | -29.121 | 32.50 | 8.999 |
| 24 | -0.898 | -0.897 | -0.897 | -27.502 | 30.64 | 7.998 |
| 25 | -0.898 | -0.896 | -0.896 | -25.879 | 28.87 | 6.997 |
| 26 | -0.898 | -0.897 | -0.897 | -24.277 | 27.07 | 5.996 |
| 27 | -0.898 | -0.896 | -0.896 | -22.691 | 25.31 | 5.000 |
| 28 | -0.898 | -0.897 | -0.897 | -21.107 | 23.52 | 3.999 |
| 29 | -0.898 | -0.896 | -0.896 | -19.512 | 21.78 | 2.998 |
| 30 | -0.898 | -0.898 | -0.898 | -17.941 | 19.98 | 1.997 |
| 31 | -0.898 | -0.897 | -0.897 | -16.377 | 18.26 | 0.996 |
| 32 | -0.898 | -0.898 | -0.898 | -14.822 | 16.51 | 0.000 |
| 33 | -0.898 | -0.897 | -0.897 | -13.277 | 14.79 | -0.996 |
| 34 | -0.898 | -0.897 | -0.897 | -11.734 | 13.08 | -1.997 |
| 35 | -0.898 | -0.897 | -0.897 | -10.191 | 11.36 | -2.998 |
| 36 | -0.898 | -0.898 | -0.898 | -8.650 | 9.63 | -3.999 |
| 37 | -0.898 | -0.898 | -0.898 | -7.127 | 7.94 | -5.000 |
| 38 | -0.898 | -0.898 | -0.898 | -5.625 | 6.26 | -5.996 |
| 39 | -0.898 | -0.898 | -0.898 | -4.111 | 4.58 | -6.997 |
| 40 | -0.898 | -0.896 | -0.896 | -2.621 | 2.92 | -7.998 |
| 41 | -0.898 | -0.897 | -0.897 | -1.143 | 1.27 | -8.999 |
| 42 | -0.898 | -0.896 | -0.896 | 0.000 | -0.00 | -9.995 |

**Table B14 Calibration table of velocity gain block Finger 1 Joint 1**

Raw File GPVCAL23
Static calibration of position & velocity gain control blocks
Finger 1, joint 2

| data set | desired input VD-FIJ2 | actual input VEL-FIJ2 | input error (calc) | gained output (calc) | gain factor (calc) | gain voltage GV-FIJ2 |
|---|---|---|---|---|---|---|
| 1 | 0.898 | 0.898 | 0.898 | 32.654 | 36.37 | 9.995 |
| 2 | 0.898 | 0.899 | 0.899 | 30.965 | 34.45 | 8.999 |
| 3 | 0.898 | 0.897 | 0.897 | 29.254 | 32.61 | 7.998 |
| 4 | 0.898 | 0.898 | 0.898 | 27.578 | 30.70 | 6.997 |
| 5 | 0.898 | 0.898 | 0.898 | 25.889 | 28.82 | 5.996 |
| 6 | 0.898 | 0.898 | 0.898 | 24.211 | 26.95 | 5.000 |
| 7 | 0.898 | 0.898 | 0.898 | 22.529 | 25.09 | 3.999 |
| 8 | 0.898 | 0.898 | 0.898 | 20.861 | 23.23 | 2.998 |
| 9 | 0.898 | 0.898 | 0.898 | 19.173 | 21.36 | 1.997 |
| 10 | 0.898 | 0.898 | 0.898 | 17.505 | 19.50 | 0.996 |
| 11 | 0.898 | 0.898 | 0.898 | 15.803 | 17.60 | 0.000 |
| 12 | 0.898 | 0.899 | 0.899 | 14.228 | 15.84 | -0.996 |
| 13 | 0.898 | 0.900 | 0.900 | 12.578 | 13.97 | -1.997 |
| 14 | 0.898 | 0.898 | 0.898 | 10.938 | 12.18 | -2.998 |
| 15 | 0.898 | 0.898 | 0.898 | 9.297 | 10.35 | -3.999 |
| 16 | 0.898 | 0.898 | 0.898 | 7.656 | 8.52 | -5.000 |
| 17 | 0.898 | 0.898 | 0.898 | 6.037 | 6.72 | -5.996 |
| 18 | 0.898 | 0.897 | 0.897 | 4.402 | 4.91 | -6.997 |
| 19 | 0.898 | 0.898 | 0.898 | 2.787 | 3.10 | -7.998 |
| 20 | 0.898 | 0.897 | 0.897 | 1.176 | 1.31 | -8.999 |
| 21 | 0.898 | 0.898 | 0.898 | 0.000 | 0.00 | -9.995 |
| 22 | -0.898 | -0.895 | -0.895 | -31.080 | 34.73 | 9.995 |
| 23 | -0.898 | -0.896 | -0.896 | -29.441 | 32.84 | 8.999 |
| 24 | -0.898 | -0.894 | -0.894 | -27.824 | 31.14 | 7.998 |
| 25 | -0.898 | -0.896 | -0.896 | -26.223 | 29.25 | 6.997 |
| 26 | -0.898 | -0.896 | -0.896 | -24.609 | 27.48 | 5.996 |
| 27 | -0.898 | -0.896 | -0.896 | -23.027 | 25.71 | 5.000 |
| 28 | -0.898 | -0.896 | -0.896 | -21.428 | 23.91 | 3.999 |
| 29 | -0.898 | -0.897 | -0.897 | -19.846 | 22.13 | 2.998 |
| 30 | -0.898 | -0.895 | -0.895 | -18.260 | 20.40 | 1.997 |
| 31 | -0.898 | -0.895 | -0.895 | -16.684 | 18.64 | 0.996 |
| 32 | -0.898 | -0.895 | -0.895 | -15.104 | 16.88 | 0.000 |
| 33 | -0.898 | -0.894 | -0.894 | -13.535 | 15.14 | -0.996 |
| 34 | -0.898 | -0.896 | -0.896 | -11.953 | 13.34 | -1.997 |
| 35 | -0.898 | -0.895 | -0.895 | -10.393 | 11.61 | -2.998 |
| 36 | -0.898 | -0.895 | -0.895 | -8.807 | 9.84 | -3.999 |
| 37 | -0.898 | -0.895 | -0.895 | -7.242 | 8.10 | -5.000 |
| 38 | -0.898 | -0.895 | -0.895 | -5.703 | 6.37 | -5.996 |
| 39 | -0.898 | -0.895 | -0.895 | -4.141 | 4.63 | -6.997 |
| 40 | -0.898 | -0.895 | -0.895 | -2.598 | 2.90 | -7.998 |
| 41 | -0.898 | -0.896 | -0.896 | -1.094 | 1.22 | -8.999 |
| 42 | -0.898 | -0.895 | -0.895 | 0.000 | -0.00 | -9.995 |

Table B15 Calibration table of velocity gain block Finger 1 Joint 2

Raw File (FVEL23)

Static calibration of position & velocity gain control blocks
Finger 1, joint 3

| data set | desired input VD-FIJ3 | actual input VEL-FIJ3 | input error (calc) | gained output (calc) | gain factor (calc) | gain voltage GV-FIJ3 |
|---|---|---|---|---|---|---|
| 1 | 0.898 | 0.898 | 0.898 | 31.562 | 35.15 | 9.995 |
| 2 | 0.898 | 0.898 | 0.898 | 29.895 | 33.27 | 8.999 |
| 3 | 0.898 | 0.897 | 0.897 | 28.236 | 31.46 | 7.998 |
| 4 | 0.898 | 0.898 | 0.898 | 26.598 | 29.60 | 6.997 |
| 5 | 0.898 | 0.898 | 0.898 | 24.957 | 27.79 | 5.996 |
| 6 | 0.898 | 0.898 | 0.898 | 23.346 | 25.90 | 5.000 |
| 7 | 0.898 | 0.898 | 0.898 | 21.715 | 24.18 | 3.997 |
| 8 | 0.898 | 0.898 | 0.898 | 20.098 | 22.37 | 2.998 |
| 9 | 0.898 | 0.898 | 0.898 | 18.482 | 20.57 | 1.997 |
| 10 | 0.898 | 0.897 | 0.897 | 16.875 | 18.80 | 0.996 |
| 11 | 0.898 | 0.898 | 0.898 | 15.285 | 17.01 | 0.000 |
| 12 | 0.898 | 0.898 | 0.898 | 13.673 | 15.25 | -0.996 |
| 13 | 0.898 | 0.898 | 0.898 | 12.107 | 13.49 | -1.997 |
| 14 | 0.898 | 0.897 | 0.897 | 10.525 | 11.73 | -2.998 |
| 15 | 0.898 | 0.898 | 0.898 | 8.938 | 9.95 | -3.999 |
| 16 | 0.898 | 0.898 | 0.898 | 7.367 | 8.20 | -5.000 |
| 17 | 0.898 | 0.899 | 0.899 | 5.805 | 6.46 | -5.996 |
| 18 | 0.898 | 0.897 | 0.897 | 4.240 | 4.73 | -6.997 |
| 19 | 0.898 | 0.898 | 0.898 | 2.701 | 3.01 | -7.998 |
| 20 | 0.898 | 0.898 | 0.898 | 1.176 | 1.31 | -8.999 |
| 21 | 0.898 | 0.898 | 0.898 | 0.000 | 0.00 | -9.995 |
| 22 | -0.898 | -0.895 | -0.895 | -30.617 | 34.23 | 9.995 |
| 23 | -0.898 | -0.895 | -0.895 | -29.041 | 32.47 | 8.999 |
| 24 | -0.898 | -0.895 | -0.895 | -27.461 | 30.70 | 7.998 |
| 25 | -0.898 | -0.894 | -0.894 | -25.906 | 28.98 | 6.997 |
| 26 | -0.898 | -0.895 | -0.895 | -24.344 | 27.21 | 5.996 |
| 27 | -0.898 | -0.896 | -0.896 | -22.793 | 25.45 | 5.000 |
| 28 | -0.898 | -0.896 | -0.896 | -21.236 | 23.71 | 3.999 |
| 29 | -0.898 | -0.894 | -0.894 | -19.680 | 22.02 | 2.998 |
| 30 | -0.898 | -0.896 | -0.896 | -18.145 | 20.26 | 1.997 |
| 31 | -0.898 | -0.897 | -0.897 | -16.590 | 18.50 | 0.996 |
| 32 | -0.898 | -0.896 | -0.896 | -15.051 | 16.79 | 0.000 |
| 33 | -0.898 | -0.896 | -0.896 | -13.512 | 15.09 | -0.996 |
| 34 | -0.898 | -0.895 | -0.895 | -11.953 | 13.36 | -1.997 |
| 35 | -0.898 | -0.896 | -0.896 | -10.410 | 11.62 | -2.998 |
| 36 | -0.898 | -0.895 | -0.895 | -8.854 | 9.89 | -3.999 |
| 37 | -0.898 | -0.896 | -0.896 | -7.320 | 8.17 | -5.000 |
| 38 | -0.898 | -0.896 | -0.896 | -5.779 | 6.45 | -5.996 |
| 39 | -0.898 | -0.896 | -0.896 | -4.223 | 4.72 | -6.997 |
| 40 | -0.898 | -0.894 | -0.894 | -2.695 | 3.01 | -7.998 |
| 41 | -0.898 | -0.895 | -0.895 | -1.172 | 1.31 | -8.999 |
| 42 | -0.898 | -0.896 | -0.896 | 0.000 | -0.00 | -9.995 |

**Table B16** Calibration table of velocity gain block Finger 1 Joint 3

Raw File F11CALA  Data Set 3
F1J1 Position Cal

Finger 1, Joint 1

| data loop | position reading (counts) POS-F1J1 | actual position (deg) ENC-F1J1 | line fit error (deg) |
|---|---|---|---|
| 1 | -2048 | -0.0 | -0.7 |
| 2 | -2048 | -0.0 | -0.7 |
| 3 | -2048 | 0.7 | 0.0 |
| 4 | -1907 | 3.4 | 0.3 |
| 5 | -1679 | 7.5 | 0.4 |
| 6 | -1471 | 11.1 | 0.4 |
| 7 | -1183 | 15.9 | 0.2 |
| 8 | -917 | 20.4 | 0.0 |
| 9 | -691 | 24.2 | -0.0 |
| 10 | -419 | 28.7 | -0.2 |
| 11 | -95 | 34.2 | -0.4 |
| 12 | 93 | 37.4 | -0.4 |
| 13 | 313 | 41.3 | -0.3 |
| 14 | 561 | 45.6 | -0.3 |
| 15 | 802 | 49.9 | -0.2 |
| 16 | 1182 | 56.8 | 0.0 |
| 17 | 1491 | 62.3 | 0.2 |
| 18 | 1889 | 69.4 | 0.5 |
| 19 | 2037 | 72.1 | 0.5 |
| 20 | 2035 | 72.0 | 0.5 |
| 21 | 1807 | 67.9 | 0.4 |
| 22 | 1429 | 61.1 | 0.0 |
| 23 | 1121 | 55.5 | -0.2 |
| 24 | 801 | 49.7 | -0.4 |
| 25 | 531 | 44.9 | -0.5 |
| 26 | 157 | 38.3 | -0.6 |
| 27 | -344 | 29.8 | -0.4 |
| 28 | -780 | 22.5 | -0.2 |
| 29 | -1313 | 13.6 | 0.2 |
| 30 | -1669 | 7.6 | 0.4 |

position = 0.017351 * (pos reading) + 36.21

Table B17 Calibration table of position sensor Finger 1 Joint 1

Raw File F11CALA  Data Set 2
F1J1 Position Cal

Finger 1, joint 1

| data loop | position reading (counts) POS-F1J1 | actual position (deg) ENC-F1J1 | line fit error (deg) |
|---|---|---|---|
| 1 | -2048 | -0.0 | -0.7 |
| 2 | -2048 | -0.0 | -0.7 |
| 3 | -2048 | -0.0 | -0.7 |
| 4 | -1962 | 2.5 | 0.2 |
| 5 | -1760 | 6.2 | 0.4 |
| 6 | -1576 | 9.3 | 0.4 |
| 7 | -1335 | 13.4 | 0.3 |
| 8 | -991 | 19.2 | 0.0 |
| 9 | -565 | 26.3 | -0.2 |
| 10 | -134 | 33.6 | -0.4 |
| 11 | 174 | 38.8 | -0.4 |
| 12 | 327 | 41.6 | -0.4 |
| 13 | 611 | 46.6 | -0.3 |
| 14 | 883 | 51.4 | -0.2 |
| 15 | 1188 | 56.9 | 0.0 |
| 16 | 1505 | 62.6 | 0.2 |
| 17 | 1821 | 68.1 | 0.3 |
| 18 | 2037 | 72.1 | 0.5 |
| 19 | 2039 | 72.1 | 0.4 |
| 20 | 1989 | 71.2 | 0.4 |
| 21 | 1760 | 67.1 | 0.3 |
| 22 | 1336 | 59.3 | -0.1 |
| 23 | 926 | 51.9 | -0.4 |
| 24 | 529 | 44.8 | -0.6 |
| 25 | 128 | 37.8 | -0.7 |
| 26 | -377 | 29.2 | -0.5 |
| 27 | -986 | 19.1 | -0.0 |
| 28 | -1464 | 11.1 | 0.2 |
| 29 | -1750 | 6.2 | 0.3 |
| 30 | -2048 | 0.3 | -0.5 |

position = 0.017345 * (pos reading) + 36.25

Table B18 Calibration table of position sensor Finger 1 Joint 1

Raw File F11CALA  Data Set 1
F1J1 Position Cal

Finger 1, Joint 1

| data loop | position reading (counts) POS-F1J1 | actual position (deg) ENC-F1J1 | line fit error (deg) |
|---|---|---|---|
| 1 | -2048 | -0.0 | -0.7 |
| 2 | -2048 | -0.0 | -0.7 |
| 3 | -2048 | 0.0 | -0.6 |
| 4 | -1945 | 2.7 | 0.2 |
| 5 | -1559 | 9.6 | 0.4 |
| 6 | -1263 | 14.6 | 0.3 |
| 7 | -893 | 20.7 | 0.0 |
| 8 | -541 | 26.6 | -0.2 |
| 9 | -221 | 32.0 | -0.3 |
| 10 | 159 | 38.5 | -0.4 |
| 11 | 439 | 43.4 | -0.3 |
| 12 | 853 | 50.8 | -0.1 |
| 13 | 1097 | 55.2 | 0.0 |
| 14 | 1371 | 60.0 | 0.1 |
| 15 | 1813 | 67.9 | 0.4 |
| 16 | 2041 | 72.1 | 0.6 |
| 17 | 2041 | 72.1 | 0.6 |
| 18 | 2005 | 71.4 | 0.5 |
| 19 | 1803 | 67.8 | 0.4 |
| 20 | 1521 | 62.7 | 0.2 |
| 21 | 1231 | 57.4 | -0.0 |
| 22 | 779 | 49.2 | -0.4 |
| 23 | 307 | 40.9 | -0.6 |
| 24 | -132 | 33.3 | -0.6 |
| 25 | -504 | 27.1 | -0.4 |
| 26 | -1053 | 17.9 | 0.0 |
| 27 | -1486 | 10.7 | 0.3 |
| 28 | -1876 | 4.0 | 0.3 |
| 29 | -2048 | -0.0 | -0.7 |
| 30 | -2048 | -0.0 | -0.7 |

position = 0.017311 * (pos reading) + 36.15

Table B19 Calibration table of position sensor Finger 1 Joint 1

Raw File F12CALF  Data Set 3
F1J2 Position Cal

Finger 1, joint 2

| data loop | position reading (counts) POS-F1J2 | actual position (deg) ENC-F1J2 | line fit error (deg) |
|---|---|---|---|
| 1 | -2048 | -0.0 | -0.3 |
| 2 | -2047 | -0.0 | -0.4 |
| 3 | -1963 | 1.8 | -0.2 |
| 4 | -1769 | 5.9 | -0.0 |
| 5 | -1555 | 10.3 | -0.0 |
| 6 | -1257 | 16.3 | -0.0 |
| 7 | -955 | 22.4 | 0.0 |
| 8 | -627 | 29.1 | 0.1 |
| 9 | -363 | 34.5 | 0.2 |
| 10 | 71 | 43.3 | 0.3 |
| 11 | 291 | 47.8 | 0.4 |
| 12 | 608 | 54.1 | 0.3 |
| 13 | 1031 | 62.4 | 0.0 |
| 14 | 1357 | 68.7 | -0.2 |
| 15 | 1837 | 78.4 | -0.2 |
| 16 | 1975 | 81.2 | -0.1 |
| 17 | 2027 | 82.4 | -0.0 |
| 18 | 1993 | 81.7 | -0.0 |
| 19 | 1677 | 75.1 | -0.3 |
| 20 | 1213 | 65.8 | -0.2 |
| 21 | 979 | 61.3 | 0.0 |
| 22 | 701 | 55.7 | 0.0 |
| 23 | 445 | 50.6 | 0.0 |
| 24 | 187 | 45.4 | 0.0 |
| 25 | -163 | 38.2 | -0.0 |
| 26 | -584 | 29.7 | -0.1 |
| 27 | -907 | 23.2 | -0.1 |
| 28 | -1357 | 14.2 | -0.0 |
| 29 | -1666 | 8.0 | -0.0 |
| 30 | -2002 | 1.1 | -0.1 |

position = 0.020130 * (pos reading) + 41.57

Table B20 Calibration table of position sensor Finger 1 Joint 2

Raw File F12CALF  Data Set 2
F1J2 Position Cal

Finger 1, Joint 2

| data loop | position reading (counts) POS-F1J2 | actual position (deg) . ENC-F1J2 | line fit error (deg) |
|---|---|---|---|
| 1 | -2048 | -0.0 | -0.4 |
| 2 | -2041 | 0.2 | -0.3 |
| 3 | -1846 | 4.3 | -0.1 |
| 4 | -1599 | 9.3 | -0.0 |
| 5 | -1319 | 15.1 | 0.0 |
| 6 | -1072 | 20.0 | 0.0 |
| 7 | -624 | 29.2 | 0.1 |
| 8 | -299 | 35.8 | 0.2 |
| 9 | -55 | 40.9 | 0.3 |
| 10 | 207 | 46.1 | 0.4 |
| 11 | 511 | 52.2 | 0.3 |
| 12 | 857 | 58.6 | 0.2 |
| 13 | 1244 | 66.6 | -0.0 |
| 14 | 1589 | 73.5 | -0.1 |
| 15 | 1889 | 79.5 | -0.2 |
| 16 | 2030 | 82.4 | -0.0 |
| 17 | 2029 | 82.4 | -0.0 |
| 18 | 1847 | 78.6 | -0.2 |
| 19 | 1591 | 73.5 | -0.2 |
| 20 | 1261 | 66.9 | -0.1 |
| 21 | 977 | 61.3 | -0.0 |
| 22 | 627 | 54.3 | 0.0 |
| 23 | 74 | 43.1 | -0.0 |
| 24 | -434 | 32.7 | -0.2 |
| 25 | -943 | 22.5 | -0.1 |
| 26 | -1346 | 14.4 | -0.0 |
| 27 | -1665 | 8.0 | -0.0 |
| 28 | -1946 | 2.2 | -0.2 |
| 29 | -2047 | 0.0 | -0.3 |
| 30 | -2047 | 0.0 | -0.3 |

position = 0.020133 * (pos reading) + 41.6

Table B21 Calibration table of position sensor Finger 1 Joint 2

Raw File F12CALF  Data Set 1
F1J2 Position Cal

Finger 1, Joint 2

| data loop | position reading (counts) POS-F1J2 | actual position (deg) ENC-F1J2 | line fit error (deg) |
|---|---|---|---|
| 1 | -2047 | -0.0 | -0.4 |
| 2 | -2047 | -0.0 | -0.4 |
| 3 | -1810 | 5.0 | -0.2 |
| 4 | -1489 | 11.6 | -0.0 |
| 5 | -1143 | 18.6 | 0.0 |
| 6 | -928 | 22.9 | 0.0 |
| 7 | -612 | 29.4 | 0.0 |
| 8 | -399 | 33.8 | 0.2 |
| 9 | -151 | 38.8 | 0.3 |
| 10 | 221 | 46.4 | 0.4 |
| 11 | 523 | 52.4 | 0.3 |
| 12 | 813 | 58.1 | 0.1 |
| 13 | 1275 | 67.1 | -0.1 |
| 14 | 1677 | 75.1 | -0.2 |
| 15 | 1989 | 81.5 | -0.1 |
| 16 | 2052 | 82.4 | -0.0 |
| 17 | 2031 | 82.4 | -0.0 |
| 18 | 1944 | 80.6 | -0.1 |
| 19 | 1475 | 71.1 | -0.2 |
| 20 | 1211 | 65.9 | -0.0 |
| 21 | 1011 | 62.0 | 0.0 |
| 22 | 812 | 58.0 | 0.0 |
| 23 | 608 | 53.9 | 0.0 |
| 24 | 329 | 48.3 | 0.0 |
| 25 | -147 | 38.6 | -0.0 |
| 26 | -671 | 27.9 | -0.1 |
| 27 | -1216 | 17.1 | -0.0 |
| 28 | -1683 | 7.6 | -0.0 |
| 29 | -2022 | 0.6 | -0.3 |
| 30 | -2048 | 0.0 | -0.3 |

position = 0.020125 * (pos reading) + 41.59

Table B22 Calibration table of position sensor Finger 1 Joint 2

Raw File P13CALA  Data Set 1
F1J3 Position Cal
first F1J3
Finger 1, joint 3

| data loop | position reading (counts) POS-F1J3 | actual position (deg) ENC-F1J3 | line fit error (deg) |
|---|---|---|---|
| 1 | -2037 | -0.0 | -0.3 |
| 2 | -2037 | -0.0 | -0.3 |
| 3 | -2037 | -0.0 | -0.3 |
| 4 | -1950 | 2.0 | -0.2 |
| 5 | -1716 | 7.4 | -0.0 |
| 6 | -1523 | 11.8 | 0.1 |
| 7 | -1214 | 18.6 | 0.1 |
| 8 | -981 | 23.8 | 0.1 |
| 9 | -577 | 32.9 | 0.2 |
| 10 | -45 | 45.0 | 0.6 |
| 11 | 355 | 53.8 | 0.5 |
| 12 | 679 | 60.7 | 0.3 |
| 13 | 1275 | 73.4 | -0.3 |
| 14 | 1808 | 85.0 | -0.5 |
| 15 | 2045 | 90.7 | -0.0 |
| 16 | 2047 | 90.7 | -0.0 |
| 17 | 1595 | 80.2 | -0.5 |
| 18 | 1033 | 68.1 | -0.2 |
| 19 | 632 | 59.6 | 0.2 |
| 20 | 273 | 51.8 | 0.3 |
| 21 | -163 | 42.1 | 0.3 |
| 22 | -636 | 31.4 | 0.0 |
| 23 | -1056 | 22.1 | 0.0 |
| 24 | -1379 | 14.9 | -0.0 |
| 25 | -1610 | 9.8 | 0.0 |
| 26 | -2009 | 0.7 | -0.2 |
| 27 | -2042 | -0.0 | -0.2 |
| 28 | -2042 | -0.0 | -0.2 |
| 29 | -2041 | -0.0 | -0.2 |
| 30 | -2041 | -0.0 | -0.2 |

position = 0.022161 * (pos reading) + 45.42

Table B23 Calibration table of position sensor Finger 1 Joint 3

Raw File F13CALA  Data Set 2
F1J3 Position Cal
Repeat
Finger 1, joint 3

| data loop | position reading (counts) POS-F1J3 | actual position (deg) ENC-F1J3 | line fit error (deg) |
|---|---|---|---|
| 1 | -2041 | -0.0 | -0.4 |
| 2 | -2040 | -0.0 | -0.4 |
| 3 | -1927 | 2.6 | -0.3 |
| 4 | -1688 | 8.1 | -0.1 |
| 5 | -1628 | 9.5 | -0.0 |
| 6 | -1434 | 13.8 | -0.0 |
| 7 | -1222 | 18.5 | 0.0 |
| 8 | -917 | 25.3 | 0.0 |
| 9 | -779 | 28.5 | 0.2 |
| 10 | -555 | 33.5 | 0.3 |
| 11 | -214 | 41.2 | 0.5 |
| 12 | 83 | 47.9 | 0.6 |
| 13 | 426 | 55.4 | 0.5 |
| 14 | 845 | 64.3 | 0.2 |
| 15 | 1276 | 73.5 | -0.2 |
| 16 | 1807 | 85.0 | -0.4 |
| 17 | 2047 | 90.7 | 0.0 |
| 18 | 2047 | 90.8 | 0.1 |
| 19 | 1945 | 88.2 | -0.2 |
| 20 | 1621 | 80.9 | -0.4 |
| 21 | 1386 | 75.8 | -0.3 |
| 22 | 1139 | 70.4 | -0.2 |
| 23 | 899 | 65.4 | 0.0 |
| 24 | 598 | 58.9 | 0.2 |
| 25 | 213 | 50.4 | 0.3 |
| 26 | -195 | 41.4 | 0.2 |
| 27 | -677 | 30.5 | -0.0 |
| 28 | -1089 | 21.4 | -0.0 |
| 29 | -1483 | 12.7 | -0.0 |
| 30 | -1828 | 4.9 | -0.2 |

position = 0.022075 * (pos reading) + 45.48

Table B24 Calibration table of position sensor Finger 1 Joint 3

Raw File P13CALA Data Set 3
F1J3 Position Cal
encoder remounted
Finger 1, joint 3

| data loop | position reading (counts) POS-F1J3 | actual position (deg) ENC-F1J3 | line fit error (deg) |
|---|---|---|---|
| 1 | -2044 | -0.0 | -0.5 |
| 2 | -2043 | -0.0 | -0.5 |
| 3 | -1933 | 2.4 | -0.6 |
| 4 | -1508 | 12.2 | -0.1 |
| 5 | -1165 | 19.9 | -0.0 |
| 6 | -821 | 27.6 | 0.0 |
| 7 | -552 | 33.8 | 0.3 |
| 8 | -322 | 39.0 | 0.5 |
| 9 | -80 | 44.6 | 0.6 |
| 10 | 152 | 49.7 | 0.6 |
| 11 | 326 | 53.4 | 0.5 |
| 12 | 612 | 59.7 | 0.5 |
| 13 | 1058 | 69.0 | -0.0 |
| 14 | 1565 | 79.8 | -0.5 |
| 15 | 1966 | 88.9 | -0.3 |
| 16 | 2038 | 90.6 | -0.1 |
| 17 | 2040 | 90.7 | -0.0 |
| 18 | 1939 | 88.2 | -0.3 |
| 19 | 1781 | 84.6 | -0.4 |
| 20 | 1485 | 78.0 | -0.5 |
| 21 | 1013 | 68.0 | -0.0 |
| 22 | 589 | 59.0 | 0.3 |
| 23 | 250 | 51.6 | 0.4 |
| 24 | -35 | 45.3 | 0.4 |
| 25 | -380 | 37.4 | 0.2 |
| 26 | -794 | 28.1 | -0.0 |
| 27 | -1034 | 22.8 | -0.0 |
| 28 | -1367 | 15.3 | -0.2 |
| 29 | -1722 | 7.3 | -0.3 |
| 30 | -2034 | 0.0 | -0.6 |

position = 0.022106 * (pos reading) + 45.68

Table B25 Calibration table of position sensor Finger 1 Joint 3

Raw File P13CALA  Data Set 4
F1J3 Position Cal
repeat
Finger 1, Joint 3

| data loop | position reading (counts) POS-F1J3 | actual position (deg) ENC-F1J3 | line fit error (deg) |
|---|---|---|---|
| 1 | -2043 | -0.0 | -0.5 |
| 2 | -2026 | 0.4 | -0.5 |
| 3 | -1505 | 12.3 | -0.0 |
| 4 | -1211 | 18.9 | -0.0 |
| 5 | -902 | 25.8 | 0.1 |
| 6 | -734 | 29.6 | 0.2 |
| 7 | -449 | 36.0 | 0.3 |
| 8 | -199 | 41.7 | 0.5 |
| 9 | 122 | 49.0 | 0.7 |
| 10 | 415 | 55.4 | 0.6 |
| 11 | 878 | 65.5 | 0.3 |
| 12 | 1374 | 75.7 | -0.3 |
| 13 | 1685 | 82.4 | -0.5 |
| 14 | 1883 | 86.9 | -0.3 |
| 15 | 2043 | 90.6 | -0.2 |
| 16 | 2043 | 90.7 | -0.0 |
| 17 | 1887 | 86.9 | -0.4 |
| 18 | 1679 | 82.3 | -0.5 |
| 19 | 1279 | 73.7 | -0.3 |
| 20 | 919 | 64.0 | 0.0 |
| 21 | 633 | 59.9 | 0.3 |
| 22 | 229 | 51.2 | 0.4 |
| 23 | -11 | 45.9 | 0.5 |
| 24 | -267 | 40.0 | 0.2 |
| 25 | -631 | 31.8 | 0.1 |
| 26 | -1061 | 22.1 | -0.0 |
| 27 | -1461 | 13.3 | -0.1 |
| 28 | -1695 | 8.0 | -0.2 |
| 29 | -1876 | 3.9 | -0.3 |
| 30 | -2038 | 0.0 | -0.5 |

position = 0.022091 * (pos reading) + 45.65

**Table B26 Calibration Table of Position Sensor Finger 1 Joint 3**

Raw File T10FCALA
F1J0 Flexion Calibration, 0-10-0 lbs
Finger 1, joint 0

| data set | flexion reading (counts) FLX-F1J0 | actual flexion (lbf) TEN-F1J0 | line fit error (lbf) |
|---|---|---|---|
| 1 | -2046 | -0.0 | 0.2 |
| 2 | -1781 | 1.7 | 0.0 |
| 3 | -1555 | 3.3 | 0.1 |
| 4 | -1341 | 4.7 | 0.1 |
| 5 | -1154 | 6.1 | 0.2 |
| 6 | -917 | 7.7 | 0.2 |
| 7 | -700 | 9.0 | -0.0 |
| 8 | -562 | 10.1 | 0.2 |
| 9 | -680 | 9.0 | -0.0 |
| 10 | -913 | 7.3 | -0.2 |
| 11 | -1155 | 5.6 | -0.2 |
| 12 | -1376 | 4.9 | -0.2 |
| 13 | -1557 | 3.0 | -0.1 |
| 14 | -1794 | 1.5 | -0.0 |
| 15 | -2047 | -0.0 | 0.2 |

flexion = 0.000608 * (flexion reading) + 13.75

Table B27 Calibration Table of Flexion Tendon Sensor Finger 1 Joint 0

Raw File T10FCALB
F1J0 Flexion Calibration, 0-20-0 lbs
Finger 1, joint 0

| data set | flexion reading (counts) FLX-F1J0 | actual flexion (lbf) TEN-F1J0 | line fit error (lbf) |
|---|---|---|---|
| 1 | -2046 | -0.0 | 0.4 |
| 2 | -1762 | 1.8 | 0.5 |
| 3 | -1544 | 3.4 | 0.5 |
| 4 | -1181 | 5.9 | 0.3 |
| 5 | -580 | 9.6 | -0.0 |
| 6 | -212 | 12.6 | 0.4 |
| 7 | 117 | 14.9 | 0.4 |
| 8 | 543 | 18.0 | 0.5 |
| 9 | 814 | 19.8 | 0.5 |
| 10 | 665 | 18.2 | -0.0 |
| 11 | 308 | 15.1 | -0.7 |
| 12 | -112 | 12.2 | -0.6 |
| 13 | -553 | 9.2 | -0.7 |
| 14 | -1006 | 6.2 | -0.4 |
| 15 | -1474 | 3.4 | -0.1 |
| 16 | -1757 | 1.8 | 0.0 |
| 17 | -2044 | -0.0 | 0.4 |

error = 0.000505 * (flexion reading) + 12.71

Table B28 Calibration Table of Flexion Tendon Sensor Finger 1 Joint 0

Raw File T1OFCALC
F1J0 Flexion Calibration, 0-30-0 lbs
Finger 1, joint 0

| data set | flexion reading (counts) FLX-F1J0 | actual flexion (lbf) TEN-F1J0 | line fit error (lbf) |
|---|---|---|---|
| 1 | -2047 | -0.0 | 0.8 |
| 2 | -1786 | 1.6 | 0.6 |
| 3 | -1550 | 3.0 | 0.5 |
| 4 | -1166 | 5.5 | 0.3 |
| 5 | -705 | 9.1 | 0.3 |
| 6 | -242 | 11.7 | 0.3 |
| 7 | 117 | 15.0 | 0.3 |
| 8 | 507 | 17.9 | 0.3 |
| 9 | 897 | 20.7 | 0.4 |
| 10 | 1355 | 24.4 | 0.7 |
| 11 | 1677 | 27.0 | 1.0 |
| 12 | 1930 | 29.2 | 1.3 |
| 13 | 1732 | 26.6 | 0.3 |
| 14 | 1533 | 24.4 | -0.6 |
| 15 | 1197 | 21.4 | -1.1 |
| 16 | 772 | 18.1 | -1.3 |
| 17 | 350 | 15.1 | -1.3 |
| 18 | -95 | 12.1 | -1.1 |
| 19 | -554 | 9.2 | -0.8 |
| 20 | -1081 | 5.8 | -0.3 |
| 21 | -1474 | 3.4 | 0.1 |
| 22 | -1728 | 1.9 | 0.4 |
| 23 | -2044 | -0.0 | 0.8 |

flexion = 0.007195 * (flexion reading) + 12.92

**Table B29 Calibration Table of Flexion Tendon Sensor Finger 1 Joint 0**

Raw File T11FCALC
F1J1 Flexion Calibration, 0-10-0 lbs
Finger 1, joint 1

| data set | flexion reading (counts) FLX-F1J1 | actual flexion (lbf) TEN-F1J1 | line fit error (lbf) |
|---|---|---|---|
| 1 | -2047 | -0.0 | 0.2 |
| 2 | -1805 | 1.5 | 0.2 |
| 3 | -1605 | 2.9 | 0.2 |
| 4 | -1371 | 4.6 | 0.2 |
| 5 | -1182 | 5.8 | 0.2 |
| 6 | -960 | 7.4 | 0.2 |
| 7 | -774 | 8.8 | 0.2 |
| 8 | -560 | 10.0 | 0.2 |
| 9 | -693 | 9.0 | -0.1 |
| 10 | -875 | 7.8 | -0.2 |
| 11 | -1057 | 6.2 | -0.4 |
| 12 | -1262 | 4.7 | -0.2 |
| 13 | -1477 | 3.1 | -0.2 |
| 14 | -1750 | 1.7 | -0.0 |
| 15 | -2046 | -0.0 | 0.2 |

flexion = 0.006959 * (flexion reading) + 13.92

Table B30 Calibration Table of Flexion Tendon Sensor Finger 1 Joint 1

Raw File T11FCALD
F1J1 Flexion Calibration, 0-20-0 lbs
Finger 1, joint 1

| data set | flexion reading (counts) FLX-F1J1 | actual flexion (lbf) TEN-F1J1 | line fit error (lbf) |
|---|---|---|---|
| 1 | -2048 | -0.0 | 0.4 |
| 2 | -1824 | 1.4 | 0.4 |
| 3 | -1599 | 2.9 | 0.3 |
| 4 | -1375 | 6.0 | 0.1 |
| 5 | -750 | 8.9 | 0.4 |
| 6 | -370 | 11.6 | 0.4 |
| 7 | 70 | 14.7 | 0.5 |
| 8 | 529 | 17.9 | 0.5 |
| 9 | 692 | 19.2 | 0.7 |
| 10 | 896 | 17.8 | 0.6 |
| 11 | 270 | 14.7 | -0.9 |
| 12 | -167 | 11.6 | -1.0 |
| 13 | -522 | 9.2 | -0.9 |
| 14 | -1035 | 6.0 | -0.5 |
| 15 | -1491 | 3.3 | -0.1 |
| 16 | -1752 | 1.7 | 0.0 |
| 17 | -2046 | -0.0 | 0.4 |

flexion = 0.006935 * (flexion reading) + 13.72

Table B31 Calibration Table of Flexion Tendon Sensor Finger 1 Joint 1

Raw File T11FCALE
F1J1 Flexion Calibration, 0-30-0 lbs
Finger 1, joint 1

| data set | flexion reading (counts) FLX-F1J1 | actual flexion (lbf) TEN-F1J1 | line fit error (lbf) |
|---|---|---|---|
| 1 | -2048 | -0.0 | 0.7 |
| 2 | -1834 | 1.4 | 0.6 |
| 3 | -1598 | 2.9 | 0.5 |
| 4 | -1164 | 5.9 | 0.4 |
| 5 | -776 | 8.7 | 0.4 |
| 6 | -371 | 11.4 | 0.3 |
| 7 | 101 | 14.6 | 0.3 |
| 8 | 439 | 17.3 | 0.4 |
| 9 | 868 | 20.6 | 0.6 |
| 10 | 1292 | 23.7 | 0.8 |
| 11 | 1650 | 26.5 | 1.1 |
| 12 | 1950 | 28.9 | 1.3 |
| 13 | 1812 | 27.1 | 0.5 |
| 14 | 1584 | 24.3 | -0.7 |
| 15 | 1240 | 21.2 | -1.4 |
| 16 | 840 | 18.2 | -1.5 |
| 17 | 354 | 14.9 | -1.4 |
| 18 | -110 | 11.8 | -1.2 |
| 19 | -502 | 9.3 | -0.9 |
| 20 | -949 | 6.5 | -0.5 |
| 21 | -1521 | 3.0 | 0.0 |
| 22 | -1718 | 1.9 | 0.3 |
| 23 | -2044 | -0.0 | 0.7 |

flexion = 0.007091 * (flexion reading) + 13.76

**Table B32 Calibration Table of Flexion Tendon Sensor Finger 1 Joint 1**

Raw File T12FCALA
F1J2 Flexion Calibration, 0-30-0 lbs
Finger 1, joint 2

| data set | flexion reading (counts) FLX-F1J2 | actual flexion (lbf) TEN-F1J2 | line fit error (lbf) |
|---|---|---|---|
| 1 | -2048 | -0.0 | 0.7 |
| 2 | -1814 | 1.5 | 0.6 |
| 3 | -1535 | 3.5 | 0.6 |
| 4 | -1178 | 5.8 | 0.4 |
| 5 | -677 | 9.3 | 0.3 |
| 6 | -410 | 11.5 | 0.7 |
| 7 | 31 | 14.7 | 0.7 |
| 8 | 418 | 17.4 | 0.7 |
| 9 | 879 | 21.0 | 1.0 |
| 10 | 1241 | 23.7 | 1.2 |
| 11 | 1597 | 26.4 | 1.3 |
| 12 | 1920 | 29.4 | 2.0 |
| 13 | 1776 | 26.9 | 0.6 |
| 14 | 1599 | 25.9 | -1.1 |
| 15 | 1312 | 23.1 | -1.9 |
| 16 | 928 | 18.3 | -2.0 |
| 17 | 469 | 15.0 | -2.0 |
| 18 | -2 | 12.1 | -1.6 |
| 19 | -440 | 9.4 | -1.2 |
| 20 | -979 | 6.3 | -0.6 |
| 21 | -1428 | 3.6 | -0.0 |
| 22 | -1766 | 1.6 | 0.4 |
| 23 | -2045 | -0.0 | 0.7 |

flexion = 0.007090 * (flexion reading) + 15.72

Table B33 Calibration Table of Flexion Tendon Sensor Finger 1 Joint 2

Raw File T13FCALA
F1J3 Flexion Calibration, 0-30-0 lbs
Finger 1, joint 3

| data set | flexion reading (counts) FLX-F1J3 | actual flexion (lbf) TEN-F1J3 | line fit error (lbf) |
|---|---|---|---|
| 1 | -2047 | -0.0 | 1.3 |
| 2 | -1805 | 1.5 | 1.2 |
| 3 | -1537 | 2.9 | 1.0 |
| 4 | -1053 | 6.0 | 0.6 |
| 5 | -613 | 9.2 | 0.5 |
| 6 | -295 | 11.6 | 0.7 |
| 7 | 147 | 14.7 | 0.7 |
| 8 | 640 | 18.1 | 0.6 |
| 9 | 984 | 20.6 | 0.7 |
| 10 | 1396 | 24.0 | 1.2 |
| 11 | 1719 | 26.8 | 1.7 |
| 12 | 1994 | 29.3 | 2.4 |
| 13 | 1862 | 26.9 | 0.9 |
| 14 | 1685 | 24.2 | -0.6 |
| 15 | 1406 | 21.0 | -1.8 |
| 16 | 1032 | 17.9 | -2.4 |
| 17 | 672 | 15.1 | -2.5 |
| 18 | 257 | 12.3 | -2.4 |
| 19 | -274 | 9.3 | -1.8 |
| 20 | -790 | 6.3 | -1.1 |
| 21 | -1374 | 3.2 | -0.0 |
| 22 | -1661 | 1.8 | 0.5 |
| 23 | -2047 | -0.0 | 1.4 |

flexion = 0.007016 * (flexion reading) + 12.97

Table B34 Calibration Table of Flexion Tendon Sensor Finger 1 Joint 3

Raw File T10ECALC
F1J0 Extension Calibration, 0-10-0 lbs
Finger 1, joint 0

| data set | extension reading (counts) EXT-F1J0 | actual extension (lbf) TEN-F1J0 | line fit error (lbf) |
|---|---|---|---|
| 1 | -2044 | -0.0 | 0.3 |
| 2 | -1803 | 1.4 | 0.3 |
| 3 | -1545 | 3.0 | 0.3 |
| 4 | -1301 | 4.3 | 0.1 |
| 5 | -1041 | 6.0 | 0.2 |
| 6 | -791 | 7.6 | 0.3 |
| 7 | -558 | 9.2 | 0.4 |
| 8 | -406 | 9.9 | 0.3 |
| 9 | -515 | 8.9 | -0.1 |
| 10 | -690 | 7.6 | -0.4 |
| 11 | -872 | 6.3 | -0.8 |
| 12 | -1142 | 4.7 | -0.5 |
| 13 | -1401 | 3.2 | -0.3 |
| 14 | -1725 | 1.4 | -0.2 |
| 15 | -2044 | -0.0 | 0.3 |

extension = 0.006140 * (extension reading) + 12.19

Table B35 Calibration Table of Extension Tendon Sensor Finger 1 Joint 0

Raw File T1OECALD
F1J0 Extension Calibration, 0-20-0 lbs
Finger 1, joint 0

| data set | extension reading (counts) EXT-F1J0 | actual extension (lbf) TEN-F1J0 | line fit error (lbf) |
|---|---|---|---|
| 1 | -2042 | -0.0 | 0.7 |
| 2 | -1794 | 1.5 | 0.6 |
| 3 | -1476 | 3.4 | 0.5 |
| 4 | -1071 | 5.8 | 0.3 |
| 5 | -586 | 9.2 | 0.5 |
| 6 | -156 | 12.3 | 0.7 |
| 7 | 149 | 14.4 | 0.9 |
| 8 | 505 | 17.5 | 1.0 |
| 9 | 914 | 19.7 | 1.2 |
| 10 | 725 | 17.3 | -0.0 |
| 11 | 502 | 14.6 | -1.2 |
| 12 | 184 | 12.3 | -1.5 |
| 13 | -254 | 9.4 | -1.5 |
| 14 | -824 | 6.1 | -1.1 |
| 15 | -1367 | 3.1 | -0.4 |
| 16 | -1688 | 1.6 | 0.0 |
| 17 | -2040 | -0.0 | 0.7 |

tension = 0.006550 * (extension reading) + 12.57

Table B36 Calibration Table of Extension Tendon Sensor Finger 1 Joint 0

Raw File T1OECALB
F1J0 Extension Calibration, 0-30-0 lbs
Finger 1, joint 0

| data set | extension reading (counts) EXT-F1J0 | actual extension (lbf) TEN-F1J0 | line fit error (lbf) |
|---|---|---|---|
| 1 | -2048 | -0.0 | 1.4 |
| 2 | -1605 | 2.7 | 1.2 |
| 3 | -1071 | 6.2 | 1.0 |
| 4 | -621 | 9.0 | 0.7 |
| 5 | -159 | 12.4 | 0.9 |
| 6 | 214 | 15.0 | 1.0 |
| 7 | 556 | 17.8 | 1.2 |
| 8 | 1079 | 21.1 | 1.2 |
| 9 | 1450 | 24.1 | 1.7 |
| 10 | 1787 | 26.9 | 2.2 |
| 11 | 2047 | 29.7 | 3.2 |
| 12 | 1536 | 27.2 | 1.4 |
| 13 | 1759 | 24.1 | -0.5 |
| 14 | 1524 | 21.2 | -1.7 |
| 15 | 1199 | 18.2 | -2.4 |
| 16 | 791 | 15.4 | -2.6 |
| 17 | 338 | 12.4 | -2.4 |
| 18 | -235 | 9.1 | -1.9 |
| 19 | -764 | 6.2 | -1.1 |
| 20 | -1342 | 3.3 | -0.0 |
| 21 | -2042 | -0.0 | 1.4 |

extension = 0.006819 * (extension reading) + 12.52

**Table B37 Calibration Table of Extension Tendon Sensor Finger 1 Joint 0**

Raw File T11ECALB
F1J1 Extension Calibration, 0-10-0 lbs
Finger 1, joint 1

| data set | extension reading (counts) EXT-F1J1 | actual extension (lbf) TEN-F1J1 | line fit error (lbf) |
|---|---|---|---|
| 1 | -2046 | -0.0 | 0.2 |
| 2 | -1746 | 1.7 | 0.1 |
| 3 | -1492 | 3.4 | 0.2 |
| 4 | -1302 | 4.6 | 0.2 |
| 5 | -1042 | 6.1 | 0.0 |
| 6 | -845 | 7.8 | 0.4 |
| 7 | -651 | 9.2 | 0.2 |
| 8 | -507 | 10.2 | 0.6 |
| 9 | -856 | 8.7 | -0.4 |
| 10 | -760 | 7.6 | -0.4 |
| 11 | -922 | 6.2 | -0.7 |
| 12 | -1172 | 4.6 | -0.6 |
| 13 | -1404 | 3.2 | -0.4 |
| 14 | -1678 | 1.9 | 0.0 |
| 15 | -2056 | -0.0 | 0.2 |

extension = 0.006682 * (extension reading) + 13.06

Table B38 Calibration Table of Extension Tendon Sensor Finger 1 Joint 1

Raw File T11ECALC
F1J1 Extension Calibration, 0-20-0 lbs
Finger 1, joint 1

| data set | extension reading (counts) EXT-F1J1 | actual extension (lbf) TEN-F1J1 | line fit error (lbf) |
|---|---|---|---|
| 1 | -2044 | -0.0 | 0.7 |
| 2 | -1787 | 1.6 | 0.6 |
| 3 | -1538 | 3.2 | 0.8 |
| 4 | -997 | 6.7 | 0.3 |
| 5 | -665 | 8.9 | 0.3 |
| 6 | -297 | 11.2 | 0.0 |
| 7 | 16[illegible] | 13.1 | 0.8 |
| 8 | 49[illegible] | 17.6 | 1.0 |
| 9 | 78[illegible] | 19.6 | 1.2 |
| 10 | 85[illegible] | 16.6 | -0.4 |
| 11 | 44[illegible] | 15.4 | -0.8 |
| 12 | 7[illegible] | 12.8 | -1.2 |
| 13 | -4[illegible] | 9.2 | -1.2 |
| 14 | -5[illegible] | 6.1 | -0.8 |
| 15 | -12[illegible] | 3.4 | -0.3 |
| 16 | -16[illegible] | 1.5 | 0.1 |
| 17 | -2044 | -0.0 | 0.7 |

tension = 0.006837 * (extension reading) + 12.20

**Table B39 Calibration Table of Extension Tendon Sensor Finger 1 Joint 1**

Raw File T11ECALD
F1J1 Extension Calibration, 0-30-0 lbs
Finger 1, joint 1

| data set | extension reading (counts) EXT-F1J1 | actual extension (lbf) TEN-F1J1 | line fit error (lbf) |
|---|---|---|---|
| 1 | -2042 | -0.0 | 1.3 |
| 2 | -1764 | 1.6 | 1.0 |
| 3 | -1492 | 3.3 | 0.8 |
| 4 | -990 | 6.4 | 0.4 |
| 5 | -637 | 9.1 | 0.5 |
| 6 | -302 | 11.6 | 0.6 |
| 7 | 122 | 14.5 | 0.6 |
| 8 | 505 | 18.1 | 0.9 |
| 9 | 909 | 21.1 | 1.0 |
| 10 | 1290 | 23.5 | 1.3 |
| 11 | 1693 | 26.5 | 1.8 |
| 12 | 1968 | 29.4 | 2.4 |
| 13 | 1902 | 27.3 | 0.7 |
| 14 | 1686 | 24.1 | -1.0 |
| 15 | 1406 | 21.2 | -1.5 |
| 16 | 1056 | 18.2 | -2.4 |
| 17 | 665 | 15.3 | -2.5 |
| 18 | 198 | 12.2 | -2.2 |
| 19 | -350 | 9.0 | -1.6 |
| 20 | -873 | 6.0 | -0.9 |
| 21 | -1254 | 3.5 | -0.0 |
| 22 | -1687 | 1.7 | 0.6 |
| 23 | -2041 | -0.0 | 1.3 |

extension = 0.007079 * (extension reading) + 13.08

**Table B40 Calibration Table of Extension Tendon Sensor Finger 1 Joint 1**

Raw File T12ECALA
F1J2 Extension Calibration, 0-10-0 lbs
Finger 1, joint 2

| data set | extension reading (counts) EXT-F1J2 | actual extension (lbf) TEN-F1J2 | line fit error (lbf) |
|---|---|---|---|
| 1 | -2047 | -0.0 | 0.3 |
| 2 | -1837 | 1.2 | 0.2 |
| 3 | -1589 | 2.9 | 0.3 |
| 4 | -1365 | 4.3 | 0.3 |
| 5 | -1136 | 5.8 | 0.3 |
| 6 | -896 | 7.4 | 0.3 |
| 7 | -695 | 9.2 | 0.3 |
| 8 | -450 | 10.3 | 0.4 |
| 9 | -578 | 8.9 | -0.2 |
| 10 | -744 | 7.6 | -0.4 |
| 11 | -960 | 6.1 | -0.6 |
| 12 | -1200 | 4.6 | -0.5 |
| 13 | -1460 | 3.1 | -0.3 |
| 14 | -1719 | 1.7 | -0.1 |
| 15 | -2046 | -0.0 | 0.3 |

extension = 0.006422 * (extension reading) + 12.81

Table B41 Calibration Table of Extension Tendon Sensor Finger 1 Joint 2

Raw File T12ECALB
F1J2 Extension Calibration, 0-20-0 lbs
Finger 1, joint 2

| data set | extension reading (counts) EX1-F1J2 | actual extension (lbf) TEN-F1J2 | line fit error (lbf) |
|---|---|---|---|
| 1 | -2046 | -0.0 | 0.6 |
| 2 | -1817 | 1.3 | 0.5 |
| 3 | -1556 | 3.0 | 0.4 |
| 4 | -1020 | 6.5 | 0.4 |
| 5 | -678 | 8.8 | 0.4 |
| 6 | -237 | 11.6 | 0.4 |
| 7 | 186 | 14.5 | 0.5 |
| 8 | 595 | 17.3 | 0.6 |
| 9 | 830 | 19.1 | 0.8 |
| 10 | 673 | 17.3 | 0.0 |
| 11 | 382 | 14.5 | -0.8 |
| 12 | 65 | 12.1 | -1.1 |
| 13 | -378 | 9.2 | -1.1 |
| 14 | -854 | 6.3 | -0.9 |
| 15 | -1400 | 3.3 | -0.7 |
| 16 | -1749 | 1.4 | 0.0 |
| 17 | -2045 | -0.0 | 0.6 |

tension = 0.006557 * (extension reading) + 12.80

Table B42 Calibration Table of Extension Tendon Sensor Finger 1 Joint 2

Raw File T12ECALC
F1J2 Extension Calibration, 0-30-0 lbs
Finger 1, joint 2

| Data set | extension reading (counts) EXT-F1J2 | actual extension (lbf) TEN-F1J2 | line fit error (lbf) |
|---|---|---|---|
| 1 | -2047 | -0.0 | 1.2 |
| 2 | -1815 | 1.3 | 1.0 |
| 3 | -1551 | 3.1 | 0.9 |
| 4 | -1072 | 6.1 | 0.6 |
| 5 | -593 | 9.2 | 0.3 |
| 6 | -223 | 11.9 | 0.4 |
| 7 | 170 | 14.7 | 0.5 |
| 8 | 574 | 17.7 | 0.6 |
| 9 | 1008 | 20.9 | 0.8 |
| 10 | 1365 | 23.6 | 1.0 |
| 11 | 1661 | 26.0 | 1.3 |
| 12 | 1988 | 29.0 | 2.1 |
| 13 | 1856 | 27.0 | 1.0 |
| 14 | 1667 | 24.1 | -0.6 |
| 15 | 1370 | 21.0 | -1.7 |
| 16 | 1057 | 18.2 | -2.0 |
| 17 | 665 | 15.5 | -2.2 |
| 18 | 180 | 12.3 | -2.0 |
| 19 | -325 | 9.3 | -1.5 |
| 20 | -805 | 6.5 | -1.0 |
| 21 | -1369 | 3.4 | -0.6 |
| 22 | -1670 | 1.8 | 0.5 |
| 23 | -2042 | -0.0 | 1.2 |

extension = 0.006995 * (extension reading) + 15.04

Table B43 Calibration Table of Extension Tendon Sensor Finger 1 Joint 2

Raw File T13ECALA
F1J3 Extension Calibration, 0-30-0 lbs
Finger 1, joint 3

| data set | extension reading (counts) EXT-F1J3 | actual extension (lbf) TEN-F1J3 | line fit error (lbf) |
|---|---|---|---|
| 1 | -2048 | -0.0 | 0.5 |
| 2 | -1834 | 1.4 | 0.4 |
| 3 | -1531 | 3.5 | 0.4 |
| 4 | -1130 | 6.2 | 0.3 |
| 5 | -657 | 9.6 | 0.3 |
| 6 | -361 | 11.6 | 0.4 |
| 7 | 12 | 14.5 | 0.4 |
| 8 | 414 | 17.3 | 0.4 |
| 9 | 905 | 20.9 | 0.5 |
| 10 | 1276 | 23.7 | 0.7 |
| 11 | 1670 | 26.8 | 0.9 |
| 12 | 1985 | 29.2 | 1.2 |
| 13 | 1780 | 26.7 | 0.0 |
| 14 | 1522 | 24.0 | -0.8 |
| 15 | 1174 | 21.2 | -1.2 |
| 16 | 746 | 18.0 | -1.3 |
| 17 | 357 | 15.4 | -1.1 |
| 18 | -184 | 11.8 | -0.9 |
| 19 | -570 | 9.3 | -0.6 |
| 20 | -1005 | 6.4 | -0.4 |
| 21 | -1457 | 3.4 | 0.0 |
| 22 | -1791 | 1.8 | 0.2 |
| 23 | -2043 | -0.0 | 0.5 |

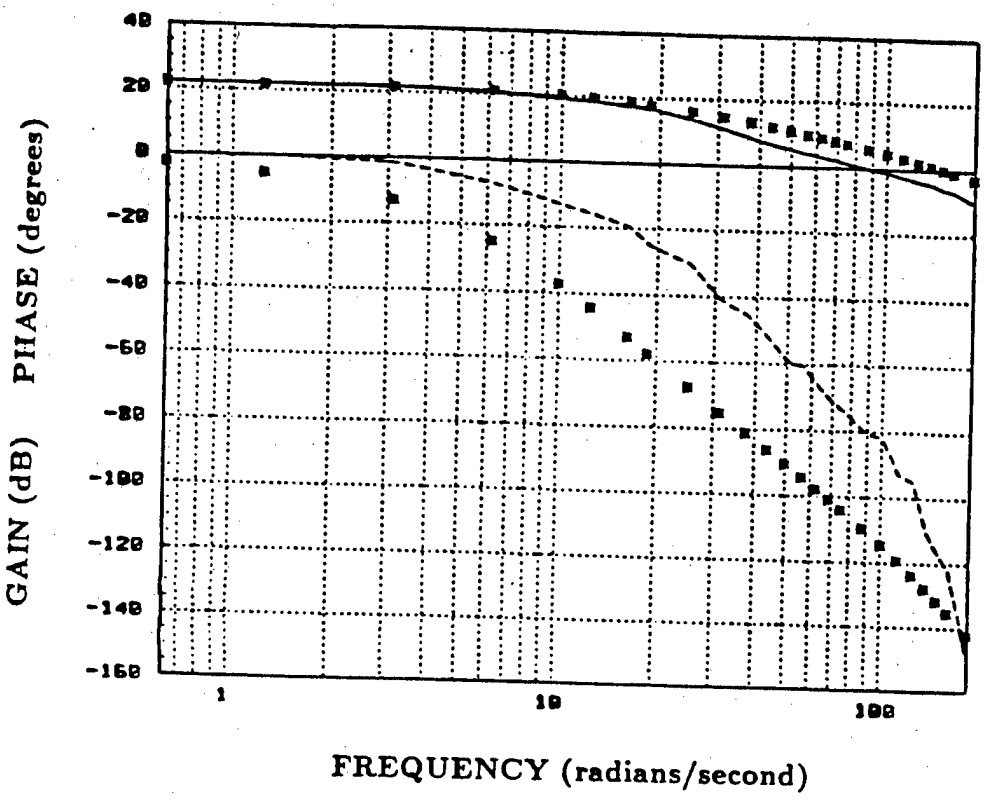
Extension = 0.007109 * (extension reading) + 15.97

Table B44 Calibration Table of Extension Tendon Sensor Finger 1 Joint 3

# APPENDIX C

# TABLES AND PLOTS OF TRANSFER FUNCTION IDENTIFICATION

This section gives the tables and plots of identification of transfer functions of joints 0, 1, 2 and 3 of finger 1 and those of coupling between joints 1, 2 and 3 of finger 1.

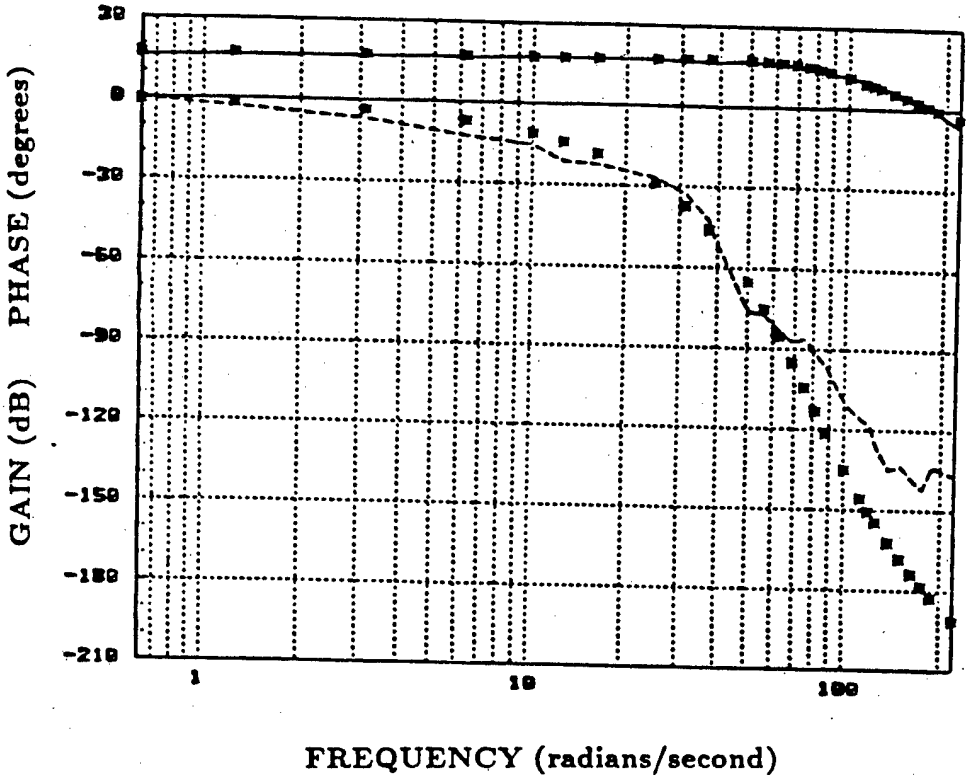



——— gain plot obtained from experimental data

- - - phase plot obtained from experimental data

* * * gain and phase plots of the estimated model

**Figure C1** Gain and phase plots of finger 1 joint 1 and its estimated model

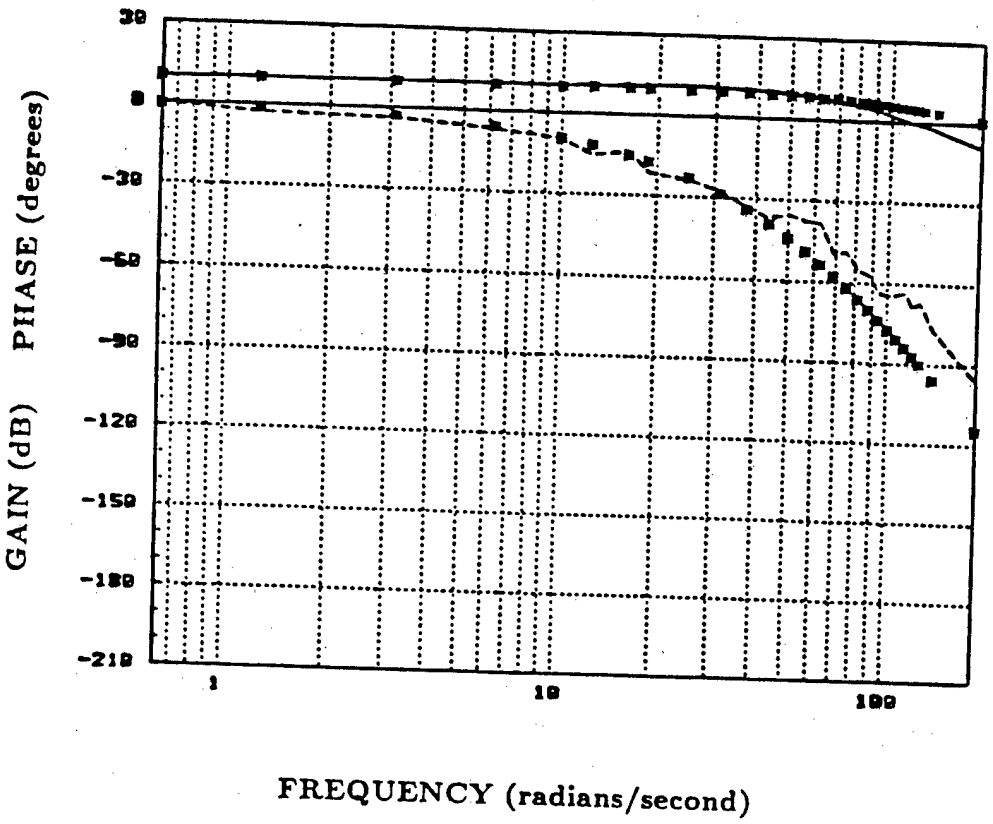



——— gain plot obtained from experimental data

- - - phase plot obtained from experimental data

* * * gain and phase plots of the estimated model

**Figure C2** **Gain and phase plots of finger 1 joint 2 and its estimated model**

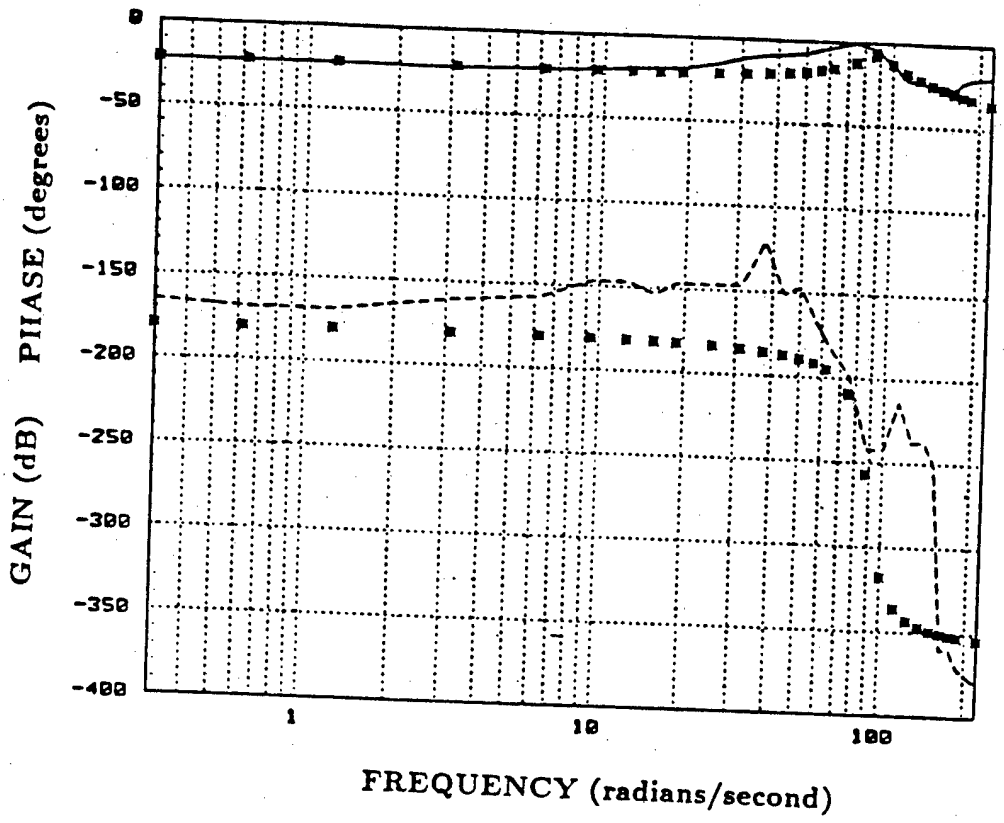



——— gain plot obtained from experimental data

- - - phase plot obtained from experimental data

* * * gain and phase plots of the estimated model

**Figure C3** **Gain and phase plots of finger 1 joint 3 and its estimated model**

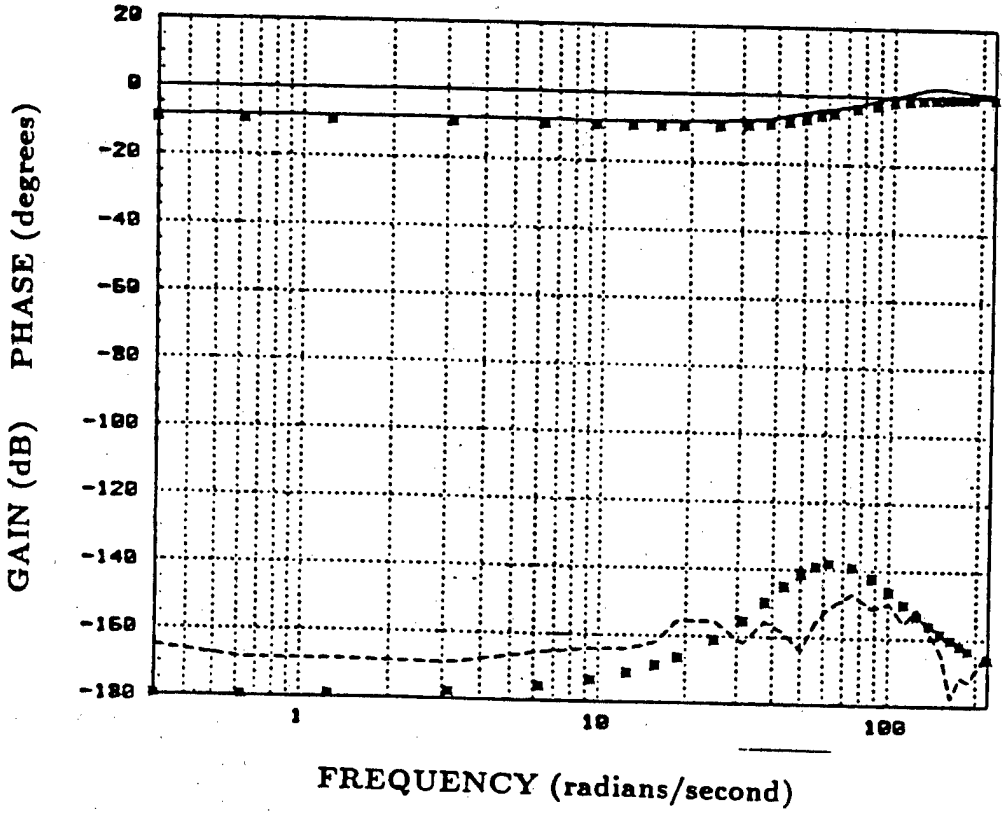



——— gain plot obtained from experimental data

- - - phase plot obtained from experimental data

* * * gain and phase plots of the estimated model

**Figure C4 Gain and phase plots of output-output coupling between joints 1 and 3 and those of their estimated model**

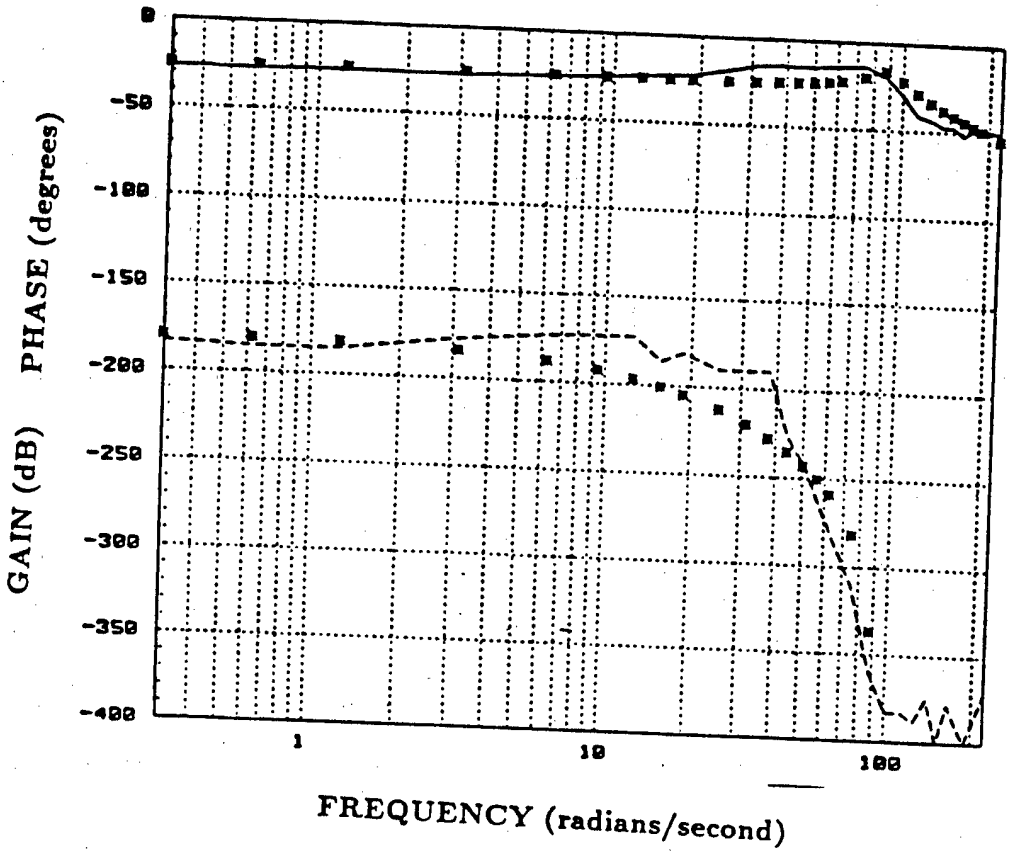



——— gain plot obtained from experimental data

- - - phase plot obtained from experimental data

* * * gain and phase plots of the estimated model

**Figure C5 Gain and phase plots of output-output coupling between joints 2 and 3 and those of their estimated model**

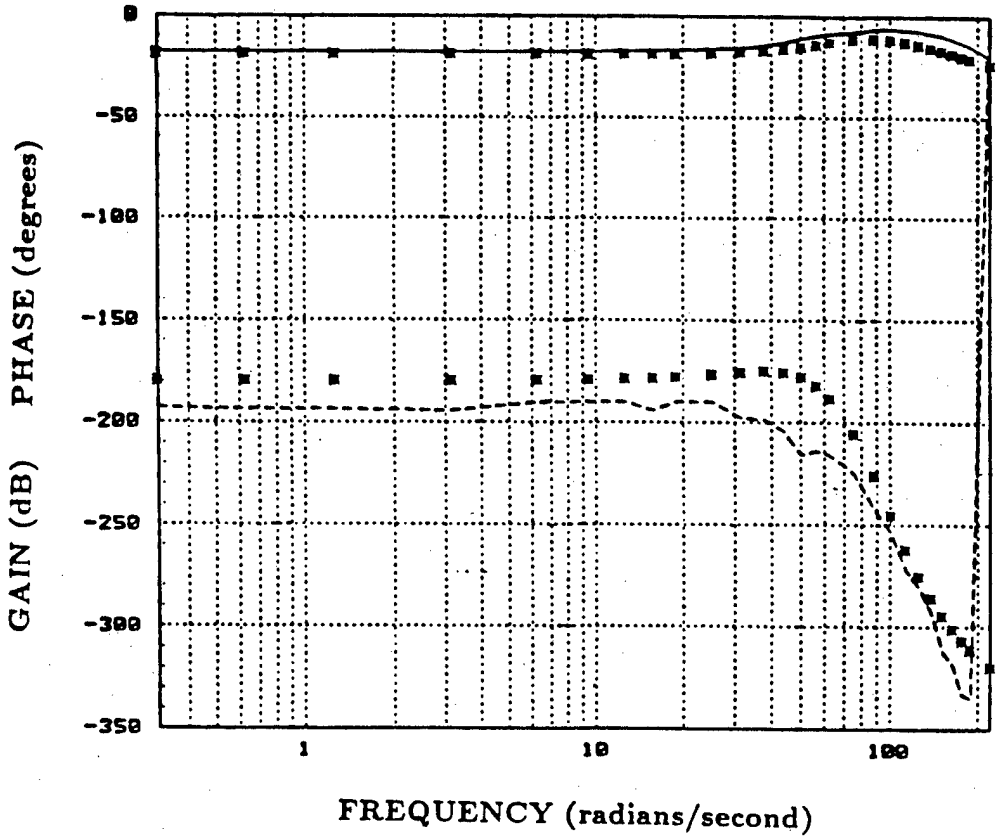



——— gain plot obtained from experimental data

- - - phase plot obtained from experimental data

* * * gain and phase plots of the estimated model

**Figure C6 Gain and phase plots of input-output coupling between joints 1 and 3 and those of their estimated model**




——— gain plot obtained from experimental data

- - - phase plot obtained from experimental data

* * * gain and phase plots of the estimated model

**Figure C7 Gain and phase plots of input-output coupling between joints 2 and 3 and those of their estimated model**

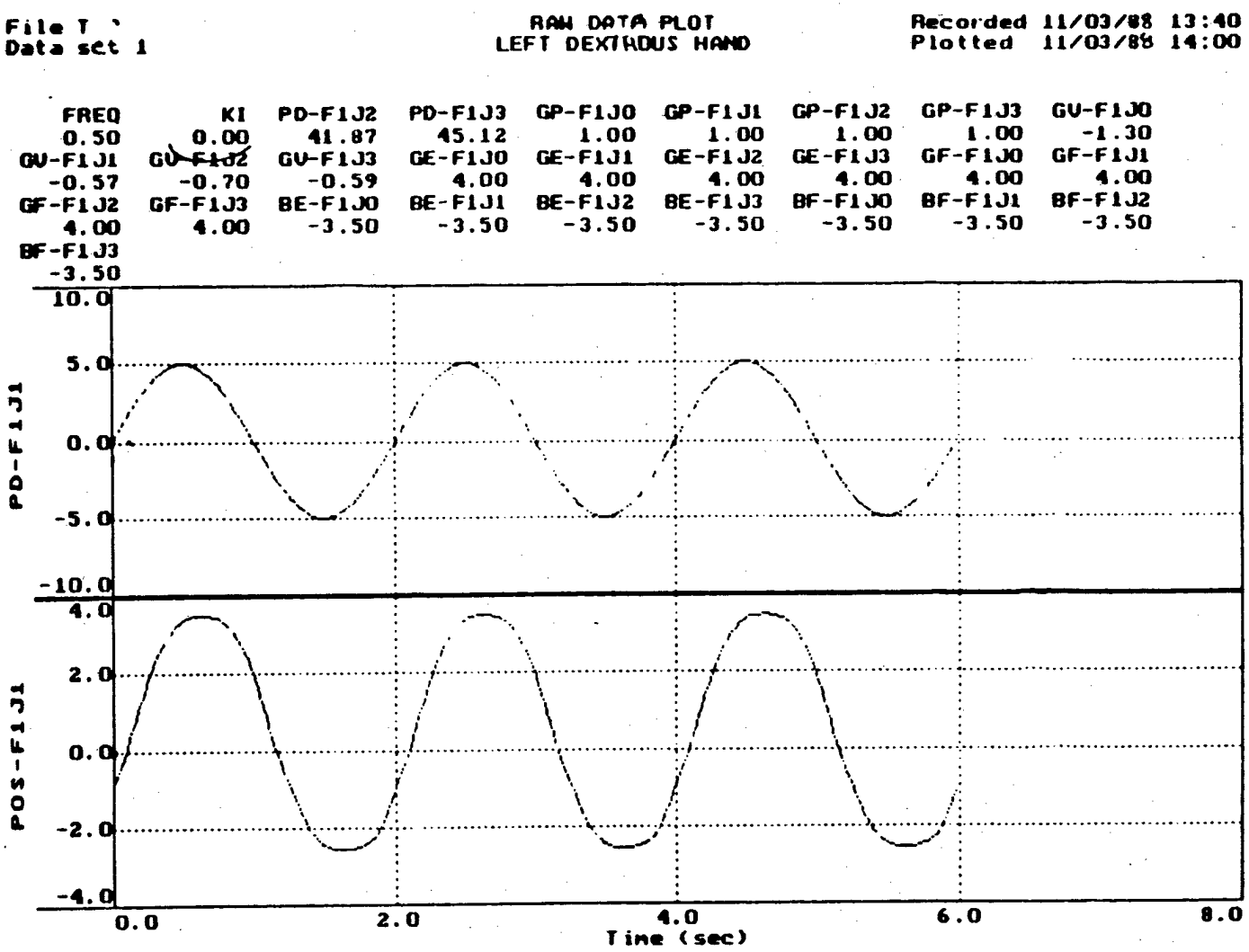



**Figure C8 Response of finger 1 joint 1 for $K_I = 0$**

DTIC COULD NOT GET MISSING

PAGE FROM CONTRIBUTOR

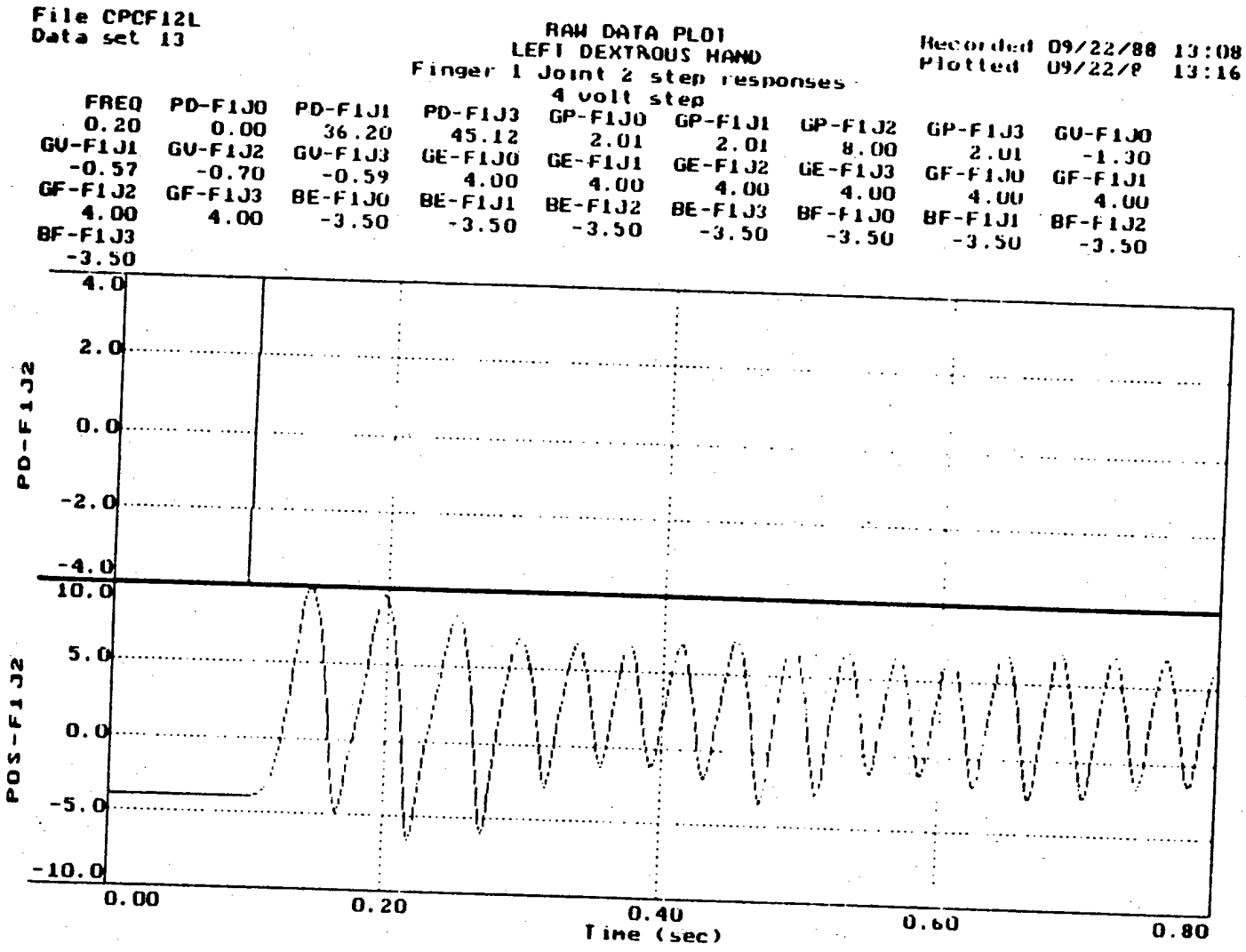



**Figure C10 Step response of finger 1 joint 2 for $K_p = 8$, $K_v = 0$**

| FREQUENCY hertz | INPUT AMPLITUDE volts | OUTPUT AMPLITUDE volts | OUTPUT / INPUT |
|---|---|---|---|
| 0.5 | 0.99 | 0.1 | 0.101 |
| 0.5 | 1.39 | 0.19 | 0.1367 |
| 0.5 | 1.79 | 0.41 | 0.2291 |
| 0.5 | 2.19 | 1.09 | 0.4977 |
| 0.5 | 2.59 | 2.82 | 1.0889 |
| 0.5 | 2.38 | 1.93 | 0.8109 |
| 0.5 | 2.99 | 5.04 | 1.6856 |
| 0.5 | 2.99 | 5.7 | 1.9064 |
| 2.0 | 0.99 | 0.1 | 0.101 |
| 2.0 | 1.39 | 0.19 | 0.1367 |
| 2.0 | 1.79 | 0.37 | 0.2067 |
| 2.0 | 2.19 | 0.87 | 0.3972 |
| 2.0 | 2.38 | 1.22 | 0.5126 |
| 2.0 | 2.59 | 1.58 | 0.61 |
| 2.0 | 2.99 | 2.44 | 0.8161 |
| 2.0 | 3.39 | 3.72 | 1.0973 |
| 2.0 | 3.79 | 4.84 | 1.277 |
| 2.0 | 4.19 | 6.2 | 1.479 |

**Table C1** Response of finger 1 joint 1 with open position loop and closed force loop

FREQUENCY RESPONSE DATA OF FINGER 1 JOINT 0

| FREQUENCY hertz | GAIN dB | PHASE deg | GAIN dB | PHASE deg |
|---|---|---|---|---|
| 0.6283 | 20.6531 | 0.0000 | 20.4933 | -2.3989 |
| 1.2566 | 20.7593 | -2.8080 | 20.4734 | -4.7909 |
| 3.1416 | 20.3483 | -4.3200 | 20.3364 | -11.8606 |
| 6.2832 | 19.0178 | -10.0800 | 19.8792 | -22.9538 |
| 10.0531 | 17.6476 | -12.9600 | 19.0579 | -34.5888 |
| 12.5664 | 16.7175 | -18.7200 | 18.4155 | -41.2246 |
| 16.2734 | 15.6923 | -22.1616 | 17.4137 | -49.5046 |
| 18.8496 | 15.0717 | -21.6000 | 16.7133 | -54.3506 |
| 25.1327 | 13.1493 | -27.3600 | 15.0752 | -63.7977 |
| 31.4159 | 10.9926 | -37.4400 | 13.5862 | -70.8855 |
| 37.6991 | 8.8671 | -43.2000 | 12.2504 | -76.4605 |
| 43.9823 | 6.5141 | -47.1600 | 11.0494 | -81.0334 |
| 50.2655 | 4.7567 | -53.2800 | 9.9622 | -84.9154 |
| 56.5487 | 3.3771 | -54.0000 | 8.9695 | -88.3023 |
| 62.8319 | 2.1940 | -59.0400 | 8.0559 | -91.3214 |
| 69.1150 | 1.3743 | -70.5600 | 7.2085 | -94.0583 |
| 75.3982 | 0.2928 | -69.1200 | 6.4173 | -96.5724 |
| 87.9646 | -1.7596 | -79.9200 | 4.9719 | -101.0892 |
| 100.5310 | -3.8937 | -87.8400 | 3.6709 | -105.0920 |
| 113.0973 | -5.8198 | -92.8800 | 2.4813 | -108.7066 |
| 125.6637 | -7.0755 | -95.0400 | 1.3803 | -112.0102 |
| 138.2301 | -9.1407 | -102.2400 | 0.3515 | -115.0535 |
| 157.0796 | -10.9205 | -114.8400 | -1.0816 | -119.2054 |

**Table C2** **Gain and phase data of finger 1 joint 0 and of its estimated model**

FREQUENCY RESPONSE DATA OF FINGER 1 JOINT 1

| FREQUENCY hertz | GAIN dB | PHASE deg. | GAIN dB | PHASE deg. |
|---|---|---|---|---|
| 0.6283 | 22.1457 | 0.0000 | 21.9305 | -2.6151 |
| 1.2566 | 22.1755 | 0.0000 | 21.9077 | -5.2219 |
| 3.1416 | 21.8684 | -1.4400 | 21.7508 | -12.9116 |
| 6.2832 | 20.7478 | -7.2000 | 21.2324 | -24.8926 |
| 10.0531 | 19.2685 | -12.9600 | 20.3175 | -37.2929 |
| 12.5664 | 18.5306 | -15.8400 | 19.6142 | -44.2820 |
| 16.2734 | 17.4431 | -19.3644 | 18.5347 | -52.9326 |
| 18.8496 | 16.7711 | -24.8400 | 17.7901 | -57.9727 |
| 25.1327 | 13.8016 | -30.2400 | 16.0728 | -67.8047 |
| 31.4159 | 11.1617 | -41.0400 | 14.5330 | -75.2524 |
| 37.6991 | 8.6563 | -45.3600 | 13.1629 | -81.2068 |
| 43.9823 | 6.8156 | -52.2000 | 11.9374 | -86.1866 |
| 50.2655 | 5.3873 | -59.0400 | 10.8311 | -90.5008 |
| 56.5487 | 4.2418 | -60.4800 | 9.8230 | -94.3410 |
| 62.8319 | 3.2612 | -66.2400 | 8.8958 | -97.8305 |
| 69.1150 | 2.3995 | -70.5600 | 8.0360 | -101.0516 |
| 75.3982 | 1.5311 | -73.4400 | 7.2329 | -104.0608 |
| 87.9646 | 0.1291 | -79.9200 | 5.7642 | -109.5930 |
| 100.5310 | -1.0717 | -82.0800 | 4.4389 | -114.6350 |
| 113.0973 | -2.3181 | -92.8800 | 3.2231 | -119.3044 |
| 125.6637 | -3.4439 | -95.0400 | 2.0935 | -123.6726 |
| 138.2301 | -4.3588 | -110.1600 | 1.0333 | -127.7863 |
| 150.7964 | -5.6576 | -116.6400 | 0.0305 | -131.6775 |
| 163.3628 | -6.6581 | -121.6800 | -0.9240 | -135.3698 |
| 188.4956 | -10.0212 | -149.0400 | -2.7137 | -142.2269 |

**Table C3** **Gain and phase data of finger 1 joint 1 and of its estimated model**

FREQUENCY RESPONSE DATA OF FINGER 1 JOINT 2

| FREQUENCY hertz | GAIN dB | PHASE deg. | GAIN dB | PHASE deg. |
|---|---|---|---|---|
| 0.6283 | 22.1457 | 0.0000 | 4.9590 | -0.6960 |
| 1.2566 | 22.1755 | 0.0000 | 4.9595 | -1.3922 |
| 3.1416 | 21.8684 | -1.4400 | 4.9626 | -3.4826 |
| 6.2832 | 20.7478 | -7.2000 | 4.9738 | -6.9808 |
| 10.0531 | 19.2685 | -12.9600 | 4.9965 | -11.2213 |
| 12.5664 | 18.5306 | -15.8400 | 5.0166 | -14.0865 |
| 16.2734 | 17.4431 | -19.3644 | 5.0523 | -18.3874 |
| 18.8496 | 16.7711 | -24.8400 | 5.0802 | -21.4401 |
| 25.1327 | 13.8016 | -30.2400 | 5.1525 | -29.1590 |
| 31.4159 | 11.1617 | -41.0400 | 5.2149 | -37.3455 |
| 37.6991 | 8.6563 | -45.3600 | 5.2419 | -46.0699 |
| 43.9823 | 6.8156 | -52.2000 | 5.2019 | -55.3461 |
| 50.2655 | 5.3873 | -59.0400 | 5.0604 | -65.1047 |
| 56.5487 | 4.2418 | -60.4800 | 4.7863 | -75.1771 |
| 62.8319 | 3.2612 | -66.2400 | 4.3603 | -85.3094 |
| 69.1150 | 2.3995 | -70.5600 | 3.7804 | -95.2095 |
| 75.3982 | 1.5311 | -73.4400 | 3.0629 | -104.6135 |
| 87.9646 | 0.1291 | -79.9200 | 1.3333 | -121.2843 |
| 100.5310 | -1.0717 | -82.0800 | -0.5846 | -134.8759 |
| 113.0973 | -2.3181 | -92.8800 | -2.5180 | -145.8021 |
| 125.6637 | -3.4439 | -95.0400 | -4.3819 | -154.6803 |
| 138.2301 | -4.3588 | -110.1600 | -6.1453 | -162.0401 |
| 150.7964 | -5.6576 | -116.6400 | -7.8031 | -168.2725 |
| 163.3628 | -6.6581 | -121.6800 | -9.3606 | -173.6531 |
| 188.4956 | -10.0212 | -149.0400 | -12.2105 | -182.5769 |

**Table C4** **Gain and phase data of finger 1 joint 2 and of its estimated model**

| FREQ(HZ) | FREQ(RAD) | GAIN(DB) | PHASE(DEG) | GAIN(DB) | PHASE(DEG) |
|---|---|---|---|---|---|
| 0.1000 | 0.6283 | 9.8535 | 0.0000 | 9.9998 | -0.5954 |
| 0.2000 | 1.2566 | 10.1417 | -2.8080 | 9.9991 | -1.1908 |
| 0.5000 | 3.1416 | 10.2058 | -3.7440 | 9.9940 | -2.9763 |
| 1.0000 | 6.2832 | 10.4437 | -6.6240 | 9.9759 | -5.9475 |
| 1.6000 | 10.0531 | 10.5194 | -9.5040 | 9.9383 | -9.4991 |
| 2.0000 | 12.5664 | 10.6215 | -15.2640 | 9.9038 | -11.8547 |
| 2.5900 | 16.2734 | 10.6336 | -13.8564 | 9.8396 | -15.3054 |
| 3.0000 | 18.8496 | 10.8714 | -21.3840 | 9.7859 | -17.6833 |
| 4.0000 | 25.1327 | 11.2288 | -23.9040 | 9.6247 | -23.3995 |
| 5.0000 | 31.4159 | 10.8138 | -27.5040 | 9.4239 | -28.9745 |
| 6.0000 | 37.6991 | 10.3270 | -33.2640 | 9.1878 | -34.3859 |
| 7.0000 | 43.9823 | 9.8718 | -38.8640 | 8.9205 | -39.6178 |
| 8.0000 | 50.2655 | 9.3556 | -35.4240 | 8.6265 | -44.6593 |
| 9.0000 | 56.5487 | 8.8668 | -37.5840 | 8.3100 | -49.5060 |
| 10.0000 | 62.8319 | 7.9588 | -38.3040 | 7.9750 | -54.1570 |
| 11.0000 | 69.1150 | 7.0747 | -49.4640 | 7.6248 | -58.6151 |
| 12.0000 | 75.3982 | 6.3218 | -48.3840 | 7.2627 | -62.8855 |
| 13.0000 | 81.6814 | 5.3769 | -55.2240 | 6.8913 | -66.9751 |
| 14.0000 | 87.9646 | 4.5011 | -57.0240 | 6.5130 | -70.8920 |
| 15.0000 | 94.2478 | 3.6271 | -63.5040 | 6.1297 | -74.6449 |
| 16.0000 | 100.5310 | 2.8795 | -64.2240 | 5.7430 | -78.2425 |
| 17.0000 | 106.8142 | 1.4430 | -64.2240 | 5.3544 | -81.6939 |
| 18.0000 | 113.0973 | 0.5541 | -63.5040 | 4.9648 | -85.0074 |
| 19.0000 | 119.3805 | -0.4264 | -68.9040 | 4.5752 | -88.1915 |
| 20.0000 | 125.6637 | -0.9908 | -67.1040 | 4.1864 | -91.2538 |
| 22.0000 | 138.2301 | -2.6562 | -77.1840 | 3.4134 | -97.0420 |
| 22.0000 | 138.2301 | -2.6562 | -77.1840 | 3.4134 | -97.0420 |
| 30.0000 | 188.4956 | -8.8798 | -95.9040 | 0.4282 | -116.5639 |

Table C5 Gain and phase data of finger 1 joint 3 and of its estimated model

COUPLING DATA OF FINGER 1 JOINT 2 DUE TO A 8V SINE INPUT AT

FINGER 1 JOINT 1

| | | ORIGINAL DATA | | FIT DATA | |
|---|---|---|---|---|---|
| FREQ(HZ) | FREQ(RAD) | GAIN(DB) | PHASE(DEG) | GAIN(DB) | PHASE(DEG) |
| 0.0500 | 0.3142 | -12.0324 | -180.0000 | -12.0436 | -179.5200 |
| 0.1000 | 0.6283 | -11.9885 | -176.3280 | -12.0430 | -179.0399 |
| 0.2000 | 1.2566 | -11.9271 | -176.3280 | -12.0406 | -178.0791 |
| 0.5000 | 3.1416 | -11.9543 | -176.9400 | -12.0234 | -175.1868 |
| 1.0000 | 6.2832 | -12.0071 | -173.8800 | -11.9574 | -170.3008 |
| 1.5000 | 9.4248 | -12.1764 | -162.7200 | -11.8325 | -165.3016 |
| 2.0000 | 12.5664 | -11.9919 | -171.3600 | -11.6302 | -160.2893 |
| 2.5000 | 15.7080 | -11.8010 | -170.8200 | -11.3325 | -155.1196 |
| 3.0000 | 18.8496 | -11.7654 | -162.7200 | -10.9282 | -150.1970 |
| 4.0000 | 25.1327 | -11.0627 | -154.0800 | -9.8164 | -141.6457 |
| 5.0000 | 31.4159 | -8.9557 | -165.1200 | -8.4408 | -135.7935 |
| 6.0000 | 37.6991 | -7.2376 | -110.0800 | -7.0202 | -132.0828 |
| 7.0000 | 43.9823 | -6.0807 | -116.2800 | -5.7167 | -132.3680 |
| 8.0000 | 50.2655 | -4.9216 | -110.8800 | -4.6057 | -133.4723 |
| 9.0000 | 56.5487 | -3.3050 | -130.3200 | -3.7003 | -135.5136 |
| 10.0000 | 62.8319 | -1.7902 | -126.7200 | -2.9814 | -137.9989 |
| 12.0000 | 75.3982 | 1.0145 | -128.1600 | -1.9792 | -143.1718 |
| 14.0000 | 87.9646 | 1.7017 | -164.1600 | -1.3675 | -147.7998 |
| 16.0000 | 100.5310 | 1.4248 | -177.1200 | -0.9852 | -151.6478 |
| 18.0000 | 113.0973 | 0.8625 | -188.6400 | -0.7372 | -154.7871 |
| 20.0000 | 125.6637 | 0.9315 | -191.5200 | -0.5700 | -157.3523 |
| 22.0000 | 138.2301 | 0.9888 | -199.4400 | -0.4530 | -159.4681 |
| 24.0000 | 150.7964 | 0.8213 | -205.9200 | -0.3685 | -161.2341 |
| 26.0000 | 163.3628 | 0.5266 | -201.6000 | -0.3056 | -162.7259 |
| 28.0000 | 175.9292 | 0.5993 | -204.4800 | -0.2577 | -164.0006 |
| 30.0000 | 188.4956 | 0.4238 | -216.7200 | -0.2203 | -165.1012 |
| 35.0000 | 219.9115 | 0.3546 | -222.1200 | -0.1585 | -167.2881 |

Table C6 Gain and phase data of coupling between finger 1 joints 1 and 2 and of its estimated model

COUPLING DATA OF FINGER 1 JOINT 3 DUE TO A SINE INPUT AT FINGER 1 JOINT 1

| FREQ hertz | FREQ rad/sec. | GAIN dB | PHASE deg. | GAIN dB | PHASE deg. |
|---|---|---|---|---|---|
| 0.0500 | 0.3142 | -22.3052 | -165.2400 | -22.0002 | -180.0400 |
| 0.1000 | 0.6283 | -22.4929 | -168.4800 | -21.9999 | -180.0800 |
| 0.2000 | 1.2566 | -22.5[illegible]8 | -168.4800 | -21.9986 | -180.1600 |
| 0.5000 | 3.1416 | -22.5631 | -160.7400 | -21.9899 | -180.4005 |
| 1.0000 | 6.2832 | -21.8616 | -157.3200 | -21.9587 | -180.8039 |
| 1.5000 | 9.4248 | -21.1264 | -146.8800 | -21.9065 | -181.2131 |
| 2.0000 | 12.5664 | -20.2001 | -146.8800 | -21.8328 | -181.6314 |
| 2.5000 | 15.7080 | -19.5261 | -153.9000 | -21.7372 | -182.0619 |
| 3.0000 | 18.8496 | -19.1250 | -146.8800 | -21.6190 | -182.5085 |
| 4.0000 | 25.1327 | -15.4415 | -146.8800 | -21.3110 | -183.4664 |
| 5.0000 | 31.4159 | -11.2646 | -146.8800 | -20.8991 | -184.5455 |
| 6.0000 | 37.6991 | -9.5118 | -120.9600 | -20.3692 | -185.8015 |
| 7.0000 | 43.9823 | -8.5795 | -151.2000 | -19.7011 | -187.3170 |
| 8.0000 | 50.2655 | -7.4045 | -146.8800 | -18.8660 | -189.2209 |
| 9.0000 | 56.5487 | -5.3029 | -165.2400 | -17.8218 | -191.7299 |
| 10.0000 | 62.8319 | -3.5992 | -180.0000 | -16.5069 | -195.2367 |
| 12.0000 | 75.3982 | -0.5828 | -198.7200 | -12.6812 | -209.3338 |
| 14.0000 | 87.9646 | -4.1605 | -241.9200 | -8.0438 | -257.1140 |
| 16.0000 | 100.5310 | -13.2552 | -241.9200 | -12.4641 | -317.9540 |
| 18.0000 | 113.0973 | -22.2789 | -213.8400 | -18.0053 | -336.5406 |
| 20.0000 | 125.6637 | -23.2001 | -237.6000 | -21.9109 | -343.6120 |
| 22.0000 | 138.2301 | -26.0824 | -237.6000 | -24.8808 | -347.2629 |
| 24.0000 | 150.7964 | -24.7637 | -250.0000 | -27.2880 | -349.4960 |
| 26.0000 | 163.3628 | -27.6042 | -360.0000 | -29.3223 | -351.0103 |
| 28.0000 | 175.9292 | -22.9226 | -360.0000 | -31.0914 | -352.1101 |
| 30.0000 | 188.4956 | -21.1598 | -370.0000 | -32.6612 | -352.9488 |
| 35.0000 | 219.9115 | -19.8454 | -370.0000 | -35.9701 | -354.3848 |

Table C7 Gain and phase data of coupling between finger 1 joints1 and 3 and of its estimated model

COUPLING DATA OF FINGER 1 JOINT 3 DUE TO A SINE INPUT AT FINGER 1 JOINT 2

| FREQ Hz. | FREQ rad/sec. | GAIN dB | PHASE degrees | GAIN dB | PHASE degrees |
|---|---|---|---|---|---|
| 0.0500 | 0.3142 | -0.3006 | -165.2400 | -9.5943 | -179.7925 |
| 0.1000 | 0.6283 | -0.2220 | -168.4000 | -9.5943 | -179.5849 |
| 0.2000 | 1.2566 | -0.1570 | -168.4000 | -9.5943 | -179.1695 |
| 0.5000 | 3.1416 | -0.1096 | -169.2000 | -9.5947 | -177.9192 |
| 1.0000 | 6.2832 | -0.2610 | -165.6000 | -9.5951 | -175.8059 |
| 1.5000 | 9.4248 | -0.3374 | -164.1600 | -9.5935 | -173.6293 |
| 2.0000 | 12.5664 | -0.2256 | -164.1600 | -9.5962 | -171.3623 |
| 2.5000 | 15.7080 | -0.0498 | -162.0000 | -9.5983 | -169.0839 |
| 3.0000 | 18.8496 | -0.0846 | -155.5200 | -9.5337 | -166.4821 |
| 4.0000 | 25.1327 | -7.0033 | -155.5200 | -9.3849 | -161.1213 |
| 5.0000 | 31.4159 | -7.5046 | -162.0000 | -9.0819 | -155.4633 |
| 6.0000 | 37.6991 | -7.2078 | -155.5200 | -8.5852 | -149.9376 |
| 7.0000 | 43.9823 | -5.9475 | -158.7600 | -7.8985 | -145.1005 |
| 8.0000 | 50.2655 | -5.1513 | -164.1600 | -7.0721 | -141.4179 |
| 9.0000 | 56.5487 | -4.1344 | -155.5200 | -6.1819 | -139.1041 |
| 10.0000 | 62.8319 | -3.7614 | -151.2000 | -5.3016 | -138.1113 |
| 12.0000 | 75.3982 | -2.6957 | -148.8000 | -3.7628 | -139.1516 |
| 14.0000 | 87.9646 | -1.1899 | -151.2000 | -2.6314 | -142.4190 |
| 16.0000 | 100.5310 | 0.0000 | -149.7600 | -1.8689 | -146.3160 |
| 18.0000 | 113.0973 | 0.8112 | -155.5200 | -1.3739 | -150.0392 |
| 20.0000 | 125.6637 | 2.0278 | -151.2000 | -1.0550 | -153.3035 |
| 22.0000 | 138.2301 | 2.5939 | -158.4000 | -0.8475 | -156.0697 |
| 24.0000 | 150.7964 | 2.6209 | -164.1600 | -0.7103 | -158.3897 |
| 26.0000 | 163.3628 | 1.9922 | -177.8400 | -0.6177 | -160.3367 |
| 28.0000 | 175.9292 | 1.8741 | -171.3600 | -0.5538 | -161.9806 |
| 30.0000 | 188.4956 | 1.4376 | -172.8000 | -0.5090 | -163.3798 |
| 35.0000 | 219.9115 | 0.9463 | -163.8000 | -0.4441 | -166.0917 |

Table C8 Gain and phase data of coupling between joints 2 and 3 of finger 1 and of its estimated model

# APPENDIX D

# PROGRAMS IN MATRIXx

This section gives the description of the file format used to translate raw data sets to a $MATRIX_x$ readable form, and an example of the same. This sections also contains program listings of two programs namely :

1. **PROGGPH**

   This program is used to find gain and phase between input and output of a joint

2. **PROGOOPC**

   This program is used to find the gain and phase of the coupling between the output of the joint commanded to move and those of the coupled joints.

# UTAH/MIT DH RAW DATA TRANSLATION FOR MATRIX-X

Given below is a description of the file format used to translate raw data sets to a Matrix-X readable form. Each file contains 1 data set. The $MATRIX_x$ file naming convention is 'rawfile.set' where 'rawfile' is the original raw data file name and 'set' is the data set number. $MATRIX_x$ files are 80 column ASCII and may be printed directly.

In the following description:

'a' refers to a character in an alphanumeric string,

'i' refers to a character in an integer string,

'f' refers to a character in a floating point (real) string.

$MATRIX_x$ file format:

| | |
|---|---|
| // aaaaaaaa | Raw data file name. |
| // a | Hand indicator (L or R). |
| // aaaaaaaaaaaa | Title (up to 80 characters). |
| // iiiii | Data set number. |
| // aaaaaaaaaaaa | Data set notes (up to 80 characters). |
| // ii/ii/ii ii:ii | Data set time stamp. |
| // ii | Number of constant parameters (1 - 34). |
| // ii | Number of variable parameters (1 - 21). |

| | |
|---|---|
| // iiiii | Number of loops. |
| DT = iii | Loop period (1 - 255 mS). |
| FREQU = fff.ff | Drive frequency (Hertz). |
| // aaaaaaaa fffffff | Second constant parameter name & value. |
| // . | |
| // . | |
| // . | |
| // aaaaaaaa fffffff | Last constant parameter name & value. |
| // aaaaaaaa | First variable parameter name. |
| // . | |
| // . | |
| // . | |
| // aaaaaaaa | Last variable parameter name. |
| XY = ... | Name of dataset variable for $MATRIX_x$. |
| (ffffff ffffff ffffff ... | First loop of variable parameter values. |
| // . | |
| // . | |
| // . | |
| ffffff ffffff ffffff...) ; | Last loop of variable parameter values. |

# EXAMPLE OF A RAW DATA FILE

```
// CPCF11F
// L
// closed position and force loops, F1J1 for a 8v sine wave
// 1
//
// 08/02/88 14:27
// 28
// 7
// 385
DT = 78;
FREQU = 0.10;

// PD-F1J0        0.00
// PD-F1J2        41.87
// PD-F1J3        45.12
// GP-F1J0        1.00
// GP-F1J1        0.75
// GP-F1J2        1.01
// GP-F1J3        1.01
// GV-F1J0        -1.29
// GV-F1J1        -0.55
```

```
// GV-F1J2          -0.69
// GV-F1J3          -0.58
// GE-F1J0          4.00
// GE-F1J1          4.00
// GE-F1J2          4.00
// GE-F1J3          4.00
// GF-F1J0          4.00
// GF-F1J1          4.00
// GF-F1J2          4.00
// GF-F1J3          4.00
// BE-F1J0          -3.50
// BE-F1J1          -3.50
// BE-F1J2          -3.50
// BE-F1J3          -3.50
// BF-F1J0          -3.50
// BF-F1J1          -3.50
// BF-F1J2          -3.50
// BF-F1J3          -3.50
// PD-F1J1
// POS-F1J1
// PGO-F1J1
// FLX-F1J1
```

// EXT-F1J1

// EGO-F1J1

// FGO-F1J1

XY = ...

| | | | | | | |
|---|---|---|---|---|---|---|
| [36.17 | 31.95 | 0.24 | 4.25 | 3.05 | 1.89 | 3.74 |
| 37.55 | 33.11 | 0.25 | 4.30 | 3.05 | 1.90 | 3.81 |
| 38.94 | 34.26 | 0.27 | 4.37 | 3.03 | 1.93 | 3.83 |
| 40.31 | 35.35 | 0.29 | 4.44 | 3.03 | 1.90 | 3.86 |
| 41.70 | 36.37 | 0.31 | 4.54 | 3.05 | 1.89 | 3.95 |
| 43.05 | 37.47 | 0.32 | 4.61 | 3.04 | 1.89 | 3.97 |
| 44.38 | 38.52 | 0.33 | 4.68 | 3.06 | 1.84 | 4.01 |
| 45.72 | 39.65 | 0.35 | 4.74 | 3.07 | 1.85 | 4.06 |
| 47.00 | 40.79 | 0.36 | 4.78 | 3.07 | 1.81 | 4.09 |
| 48.28 | 41.94 | 0.37 | 4.80 | 3.09 | 1.76 | 4.12 |
| 49.53 | 43.07 | 0.37 | 4.82 | 3.08 | 1.80 | 4.16 |
| 50.74 | 44.14 | 0.38 | 4.88 | 3.06 | 1.86 | 4.17 |
| 51.92 | 44.99 | 0.40 | 4.96 | 3.03 | 1.95 | 4.22 |
| 53.05 | 45.79 | 0.41 | 5.04 | 3.02 | 1.95 | 4.28 |
| 54.16 | 46.76 | 0.42 | 5.09 | 3.06 | 1.86 | 4.28 |
| 55.22 | 47.82 | 0.42 | 5.09 | 3.08 | 1.82 | 4.27 |
| 56.22 | 48.91 | 0.41 | 5.07 | 3.09 | 1.76 | 4.25 |
| 57.18 | 49.96 | 0.41 | 5.04 | 3.09 | 1.76 | 4.24 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 32.54 | 28.78 | 0.21 | 4.15 | 3.02 | 1.94 | 3.60 |
| 33.93 | 29.94 | 0.23 | 4.20 | 3.02 | 1.95 | 3.66 |
| 35.32 | 31.14 | 0.24 | 4.25 | 3.03 | 1.92 | 3.73 ]; |

```
////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////////
//////
//////                    PROGGPH
////// The function of this program is used to find the gain and phase
////// between the input and output of a joint
//////
////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////////
//
////////////////////////////////////////////////////////////////////////
//
//    This part of the program is used to find the gain in decibels (dB)
//
////////////////////////////////////////////////////////////////////////
//
//
dtl = dt * .001;
// dt is the sampling interval of data collection. dt is specified in the
// raw data file as n where n is the sampling interval in milliseconds.
// Hence, it is multiplied by .001
frequ,
// frequ denotes the frequency of the sine wave input
pd = xy (:,1);
// xy is a matrix which has as its columns, the sampled data of different
// parameters being measured.
// pd is position desired i.e. input to the system
rs = size (pd);
r = rs (1);
// rs is a 2 × 1 matrix whose (1,1) element denotes the no. of rows of matrix pd
// element (2,1) denotes the no. of columns in matrix pd
for mn = 1 : r-1,. . .
    df (mn) = [abs (pd (mn)-pd (mn+1))];. . .
end,
// df is a column matrix of the differences between successive readings
// of the the pd matrix which is the matrix of the desired position
diffpd = max (df);
// diffpd has the maximum difference value of pd (input function) matrix, which
// is the desired position
pdl = pd (round (r-0.75 * r) : r);
//pdl is a column matrix which has the last 1/4th values of matrix pd
//
pos = xy (:,2);
```

```
// pos is a column matrix containing the sampled values of the output
for op = 1 : r-1,. . .
    dfpos (op) = [abs (pos (op)-pos (op+1))];. . .
end,
diffpos = max (dfpos);
// diffpos has the maximum difference value of the actual position matrix, pos
posl = pos (round (r-0.75 * r) : r);
maxpdl = max (pdl);
minpdl = min (pdl);
maxposl = max (posl);
minposl = min (posl);
magpdl = maxpdl - minpdl,
// magpdl returns the magnitude of input
magposl = maxposl - minposl,
// magposl returns the magnitude of output
gain = magposl/magpdl;
dbgain = 20 * log (gain)/log (10);
// db gain returns the value of the gain (in decibels)
refpdl = maxpdl - (magpdl/2);
// refpdl returns the reference line of the input function
//
//
////////////////////////////////////////////////////////////////////////////
/////
/////     This part of the program is used to find the phase shift
/////      in degrees between input and output
/////
////////////////////////////////////////////////////////////////////////////
//
q = 0;
for s = round (r-0.75 * r) : r-1,. . .
    if q = 0,. . .
        if pd (s) < pd (s+1),. . .
            if abs (pd (s)-refpdl) < 2 * diffpd/3,. . .
                q = s;. . .
                tin = dtl * s;. . .
            end,. . .
        end,. . .
    end,. . .
end,
// tin returns the time at which the input function crosses the reference value
refposl = maxposl- (magposl/2);
// refposl returns the reference value of the actual position
```

```
qo = 0;
for st = q : r-1,. . .
    if qo = 0,. . .
        if pos (st) < pos (st+1),. . .
            if abs (pos (st)-refposl) < 2 * diffpos/3,. . .
                tout = st * dtl;. . .
                qo = st;. . .
            end,. . .
        end,. . .
    end,. . .
end,
tp = 1/frequ;
// tp returns the sampling time in seconds
phase = - ( (tout - tin)/tp) * 360;
// phase returns the phase shift between input and output functions in degrees
```

```
////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////
/////
/////                    PROGOOPC
///// This program is used to find the coupling between
///// the output of joint 1 and outputs of joints 2 and 3
////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////
//
//
// This program is used to find the coupling between the outputs of joint 1
// and joint 2, and between the outputs of joint 1 and joint 3 when joint 1 is
// input with a sinusoid.
//
//
////////////////////////////////////////////////////////////////////////
//
// This part of the program is used to find the gain in decibels (dB) of the
// coupling between joints
////////////////////////////////////////////////////////////////////////
//
dtl = dt * .001;
// dt is the sampling interval of data collection. dt is specified in the raw data
// file as n where n is the sampling interval in milliseconds. Hence, it is multipied by
.001 frequ,
// frequ denotes the frequency of the sine wave input pos1 = xy (:,2); // xy is a
matrix which has as its columns, the sampled data of different parametrers // being
measured.
rs = size (pos1);
r = rs (1);
df = 0;
for mn = 1 : r-1,...
    df = [df;abs (pos1 (mn)-pos1 (mn+1))];...
end,
// df is the column matrix of the differences between succesive readings
// of the desired position
diffpos1 = max (df);
// diffpos1 is the value of the maximum difference in the desired position
pos11 = pos1 (round (r-0.65 * r) : r);
//
//
dfpos2 = 0;
dfpos3 = 0;
```

```
pos2 = xy (:,3);
pos3 = xy (:,4);
//
for op = 1 : r-1,...
    dfpos2 = [dfpos2;abs (pos2 (op)-pos2 (op+1))];...
end,
diffpos2 = max (dfpos2);
// diffpos2 is the value of the maximum difference in the position output
// of joint 2
//
for op = 1 : r-1,...
    dfpos3 = [dfpos3;abs (pos3 (op)-pos3 (op+1))];...
end,
diffpos3 = max (dfpos3);
// diffpos3 is the value of the maximum difference in the position output
// of joint 3
posl2 = pos2 (round (r-0.65 * r) : r);
posl3 = pos3 (round (r-0.65 * r) : r);
maxposl1 = max (posl1);
minposl1 = min (posl1);
maxposl2 = max (posl2);
minposl2 = min (posl2);
maxposl3 = max (posl3);
minposl3 = min (posl3);
//
magposl1 = maxposl1 - minposl1;
//
magposl2 = maxposl2 - minposl2;
gain2 = magposl2/magposl1;
dbgain2 = 20 * log (gain2)/log (10),
//
magposl3 = maxposl3 - minposl3;
gain3 = magposl3/magposl1;
dbgain3 = 20 * log (gain3)/log (10),
//
//
// the following part is used to find the phase shift
refposl = maxposl1- (magposl1/2);
q = 0;
for s = round (r-0.8 * r) : r-1,...
    if q = 0,...
        if pos (s) < pos (s+1),...
            if abs (pos (s)-refposl1) < 2 * diffpos/3,...
```

```
                q = s;...
                tin = dtl * s;...
            end,...
        end,...
    end,...
end,
//
refposl2 = maxposl2- (magposl2/2);
qo2 = 0;
for st2 = q : r-1,...
    if qo2 = 0,...
        if pos2 (st2) < pos2 (st2+1),...
            if abs (pos2 (st2)-refposl2) < 2 * diffpos2/3,...
                tout2 = st2 * dtl;...
                qo2 = st2;...
            end,...
        end,...
    end,...
end,
tp = 1/frequ;
phase2 = - ( (tout2 - tin)/tp) * 360,
// phase2 is the phase shift between the outputs of joints 1 and 2
//
refposl3 = maxposl3- (magposl3/2);
qo3 = 0;
for st3 = q : r-1,...
    if qo3 = 0,...
        if pos3 (st3) < pos3 (st3+1),...
            if abs (pos3 (st3)-refposl3) < 2 * diffpos3/3,...
                tout3 = st3 * dtl;...
                qo3 = st3;...
            end,...
        end,...
    end,...
end,
phase3 = - ((tout3 - tin)/tp) * 360,
// phase3 is the phase shift between the outputs of joints 1 and 3
table = [table;frequ 2 * pi * frequ dbgain2 phase2 dbgain3 phase3 ];
save 'TABLEPCOO11B8' table;
// saves variable table in the file TABLEPCOO11B8
```

# APPENDIX E

# C PROGRAM LISTINGS

This section gives the program listings of

1. **PROGINTEGRAL** This program is used to add an integral term to the PD controller
2. **PROGDECOUPLE** The function of this program is to implement the decoupling terms

# DIFFERENCE EQUATION DERIVATION

The derivation of the difference equation to implement the integral term, which needs to be added to the analog servo controller, is explained below. The addition of the integral term will convert the PD controller to a PID controller. The integral term can be represented as

$$\frac{\mathrm{Y(s)}}{\mathrm{X(s)}} = \frac{K_I}{\mathrm{s}}$$

where

$K_I$ – gain factor of the integral term

The above equation is converted to an equation in z-domain using bi-linear transformation $s = \frac{2\ (\mathrm{z-1})}{\mathrm{T}\ (\mathrm{z+1})}$

which results in the following equation

$$\frac{\mathrm{Y(z)}}{\mathrm{X(z)}} = \frac{K_I T z + K_I T}{2\mathrm{z-2}}$$

The difference equation was derived which is given below

$$y(k) = \frac{K_I T}{2} x(k) + \frac{K_I T}{2} x(k-1) + y(k-1)$$

A program was written in C to implement the difference equation which is given below

```
/*************************************************************
*****
*****                    PROGINTEGRAL
***** This program is used to add an integral term to the PD controller
***** This program reads the input data from a file named input.dat
***** and writes the result into a file named output.dat
*****
*************************************************************/


#include <stdio.h>

#define KI 1
#define T 0.1

main()
   {
     float x_last, y_last, x_now, y_now, cons, T1;
     FILE *infile,*outfile;

     if((infile = fopen("input.dat","r")) == NULL)
        {
          perror("input.dat");
          exit(0);
        }
     if((outfile = fopen("output.dat","w")) == NULL)
        {
          perror("output.dat");
          exit(0);
        }

     x_last = 0;
     y_last = 0;
     cons = KI * T/2;

     fscanf(infile, "%f", &x_now);
     y_now = cons * x_now + cons * x_last + y_last;
     x_last = x_now;
     T1 = T;
     fprintf(outfile,"T1\t\tXk\t\tYk\n");
     fprintf(outfile,"%f\t%f\t%f\n", T1, x_now, y_now);

     while(fscanf(infile, "%f", &x_now) != EOF)
        {
```

```
        T1 = T1 + T;
        y_last = y_now;
        y_now = cons * x_now + cons * x_last + y_last;
        fprintf(outfile,"%f\t%f\t%f\n", T1, x_now, y_now);
        x_last = x_now;
      }
  fclose(infile);
}
```

# DIFFERENCE EQUATION DERIVATION

The decoupling terms $T_{yy}^{-1}T_{xy}$ in the unreduced form result in being fourth order systems of the general form given below

$$T_{yy}^{-1}T_{xy}(s) = \frac{As^4 + Bs^3 + Cs^2 + Ds + E}{Fs^4 + Gs^3 + Hs^2 + Is + J}$$

If a reduced order equation is used, the correponding terms in the equation given above can be set to zero. Using bi-linear transformation $s = \frac{2(z-1)}{T(z+1)}$, the s-domain equation is converted to an equation in z-domain which is given below

$$\frac{Y(z)}{X(z)} = \frac{AAz^4 + BBz^3 + CCz^2 + DDz + EE}{FFz^4 + GGz^3 + HHz^2 + IIz + JJ}$$

where

$$AA = 16A + 8BT + 4CT^2 + 2DT^3 + ET^4$$

$$BB = -64B - 16BT + 4DT^3 + 4ET^4$$

$$CC = 96A - 8CT^2 + 6ET^4$$

$$DD = -64A + 16BT - 4DT^3 + 4ET^4$$

$$EE = 16A - 8BT + 4CT^2 - 2DT^3 + JT^4$$

$$FF = 16F + 8GT + 4HT^2 + 2IT^3 + JT^4$$

$$GG = -64G - 16GT + 4IT^3 + 4JT^4$$

$$HH = 96F - 8HT^2 + 6JT^4$$

$$II = -64F + 16GT - 4IT^3 + 4JT^4$$

$$JJ = 16F - 8GT + 4HT^2 + 2IT^3 + JT^4$$

The z-domain equation is converted to a a difference equation which is given below

$$y(k) = \frac{AA}{FF}x(k) + \frac{BB}{FF}x(k-1) + \frac{CC}{FF}x(k-2) + \frac{DD}{FF}x(k-3)$$
$$+ \frac{EE}{FF}x(k-4) - \frac{GG}{FF}y(k-1) - \frac{HH}{FF}y(k-2) - \frac{II}{FF}y(k-3) - \frac{JJ}{FF}y(k-4)$$

A code was written in 'C' for the difference equation which is given below. This program uses $TO_{12}$ (5.12) as an example.

```
/*****************************************************************
*****
*****                    PROGDECOUPLE
***** The function of this program is to implement the decoupling terms
*****
*****************************************************************/

#include <stdio.h>

#define A 0
#define B 0
#define C 1
#define D 48
#define E 900
#define F 0
#define G 0
#define H 1
#define I 96
#define J 3600
#define T 0.1

main()
   {
     float x_last1, x_last2,x_last3, x_last3, x_last4, y_last1, y_last2;
     float y_last3, y_last4, x_now, y_now, T1;
     FILE *infile, *outfile;

     if((infile = fopen("input.dat","r")) == NULL)
        {
          perror("input.dat");
          exit(0);
        }

     if((outfile = fopen("output.dat","w")) == NULL)
        {
          perror("output.dat");
          exit(0);
        }

     x_last1 = 0;
     x_last2 = 0;
     x_last3 = 0;
     x_last4 = 0;
```

```
y_last1 = 0;
y_last2 = 0;
y_last3 = 0;
y_last4 = 0;

AA = 16 * A + 8 * B * T + 4 * C * T * T + 2 * D * T * T * T
     + E * T * T * T * T * T;
BB = -64 * A - 16 * B * T + 4 * D * T * T * T
     + 4 * E * T * T * T * T * T;
CC = 96 * A - 8 * C * T * T
     + 6 * E * T * T * T * T * T;
DD = -64 * A + 16 * B * T - 4 * D * T * T * T
     + 4 * E * T * T * T * T * T;
EE = 16 * A - 8 * B * T + 4 * C * T * T - 2 * D * T * T * T
     + E * T * T * T * T * T;
FF = 16 * F + 8 * G * T + 4 * H * T * T + 2 * I * T * T * T
     + J * T * T * T * T * T;
GG = -64 * F - 16 * G * T + 4 * I * T * T * T
     + 4 * J * T * T * T * T * T;
HH = 96 * F - 8 * H * T * T
     + 6 * J * T * T * T * T * T;
II = -64 * F + 16 * G * T - 4 * I * T * T * T
     + 4 * J * T * T * T * T * T;
JJ = 16 * F - 8 * G * T + 4 * H * T * T - 2 * I * T * T * T
     + J * T * T * T * T * T;

fscanf(infile, "%f", &x_now);
y_now = (AA/FF) * x_now + (BB/FF) * x_last_1 + (CC/FF) * x_last2
        + (DD/FF) * x_last3 + (EE/FF) * x_last4 - (GG/FF) * y_last1
        -(HH/FF) * y_last2 -(II/FF) * y_last3 -(JJ/FF) * y_last4;
T1 = T;
fprintf(outfile,"T1\t\t\tXk\t\tYk\n");
fprintf(outfile,"%f\t%f\t%f\n", T1, x_now, y_now);

   x_last1 = x_now;
   x_last2 = x_last1;
   x_last3 = x_last2;
   x_last4 = x_last3;
   y_last1 = y_now;
   y_last2 = y_last1;
   y_last3 = y_last2;
   y_last4 = y_last3;
```

```
    while (fscanf(infile,"%f", &x_now) != EOF)
      {
        T1 = T1 + T;
        y_now = (AA/FF) * x_now + (BB/FF) * x_last_1
                + (CC/FF) * x_last2 + (DD/FF) * x_last3
                + (EE/FF) * x_last4 - (GG/FF) * y_last1
                - (HH/FF) * y_last2 - (II/FF) * y_last3
                - (JJ/FF) * y_last4;
        fprintf(outfile,"%f\t%f\t%f\t\n", T1, x_now, y_now);
        x_last1 = x_now;
        x_last2 = x_last1;
        x_last3 = x_last2;
        x_last4 = x_last3;
        y_last1 = y_now;
        y_last2 = y_last1;
      }
    fclose(infile);
    fclose(outfile);
}
```

# APPENDIX F

# GAIN BLOCK CHECK

The gain blocks in the analog controller do not produce very accurate gains. This section gives data of the gains produced by the position gain block. The notations used and their explanations are given below

| | | |
|---|---|---|
| PD | - | position desired (in degrees), first input to gain block |
| POS | - | actual position (in degrees), second input to gain block |
| PGOACT | - | actual output of the gain block = 0.1875 (PD - POS) |
| PGOCALC | - | calculated output of the gain block which is obtained by multiplying the difference between PD and POS by 0.1875 |
| GAINCALC | - | calculated gain of the gain block which is obtained by dividing PGO by the difference between PD and POS |

It can be seen from the column GAINCALC that the gains are very different from .1875 which is the gain to which the gain block has been set. Thus, it has been proved that the gains produced by the different gain blocks in the analog controller are not accurate.

| PD | POS | PGOACT | PGOCALC | GAINCALC |
|---|---|---|---|---|
| 36.1700 | 31.7600 | 0.2500 | 0.8269 | 0.0567 |
| 37.5500 | 32.9400 | 0.2700 | 0.8644 | 0.0586 |
| 38.9400 | 34.1000 | 0.2800 | 0.9075 | 0.0579 |
| 40.3100 | 35.2100 | 0.2900 | 0.9562 | 0.0569 |
| 41.7000 | 36.2700 | 0.3100 | 1.0181 | 0.0571 |
| 43.0500 | 37.3100 | 0.3300 | 1.0763 | 0.0575 |
| 44.3800 | 38.4000 | 0.3500 | 1.1212 | 0.0585 |
| 45.7200 | 39.5300 | 0.3500 | 1.1606 | 0.0565 |
| 47.0000 | 40.6700 | 0.3600 | 1.1869 | 0.0569 |
| 48.2800 | 41.7800 | 0.3700 | 1.2188 | 0.0569 |
| 49.5300 | 42.9300 | 0.3800 | 1.2375 | 0.0576 |
| 50.7400 | 44.0400 | 0.3800 | 1.2563 | 0.0567 |
| 51.9200 | 44.9900 | 0.4000 | 1.2994 | 0.0577 |
| 53.0500 | 45.8000 | 0.4100 | 1.3594 | 0.0566 |
| 54.1600 | 46.7100 | 0.4200 | 1.3969 | 0.0564 |
| 55.2200 | 47.6800 | 0.4200 | 1.4138 | 0.0557 |
| 56.2200 | 48.7500 | 0.4200 | 1.4006 | 0.0562 |
| 57.1800 | 49.7600 | 0.4200 | 1.3912 | 0.0566 |
| 58.1000 | 50.7400 | 0.4200 | 1.3800 | 0.0571 |
| 58.9400 | 51.6300 | 0.4200 | 1.3706 | 0.0575 |

| PD | POS | PGOACT | PGOCALC | GAINCALC |
|---|---|---|---|---|
| 59.7600 | 52.4100 | 0.4200 | 1.3781 | 0.0571 |
| 60.5100 | 53.1200 | 0.4200 | 1.3856 | 0.0568 |
| 61.2000 | 53.8000 | 0.4200 | 1.3875 | 0.0568 |
| 61.8200 | 54.4400 | 0.4200 | 1.3837 | 0.0569 |
| 62.3900 | 55.0300 | 0.4200 | 1.3800 | 0.0571 |
| 62.9000 | 55.5300 | 0.4200 | 1.3819 | 0.0570 |

# APPENDIX G

# SIMULATION PROGRAMS

This section gives the program listings of two programs namely **CRPROGBLS** and **MODPROGBLS**. This section also gives the figures of simulation of the decoupling algorithm in $\text{MATRIX}_x$.

```
////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////
/////
/////                    PROGRAM CRPROGBLS
///// The objective of this program is to simulate the servo loops of
///// joint 1, joint 2 and joint 3 and the decoupling scheme using
///// SYSTEM_BUILD in MATRIX_x
/////
////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////
//
//
// The blocks of super-block BLOCK1 are defined in
// the following section
kp11 = 8;
// proportional gain of PID controller of finger 1 joint 1
kv11 = 2;
// velocity gain of PID controller of finger 1 joint 1
ki11 = 0;
// integral gain of PID controller of finger 1 joint 1
kf11 = 1/124.31;
gain of the differentiator of finger 1 joint 1
n11 = 3450 * 12.5/4;
nm11 = 1;
num11 = convolve (n11,nm11);
// numerator of plant with velocity feedback
dimnum11 = size (num11);
sznum11 = dimnum11 (2) - 1;
// order of numerator of joint 1
d11 = [1 245 3450];
dn11 = 1;
dne11 = convolve (d11,dn11);
szdne11 = size (dne11);
sz2dne11 = szdne11 (2) - 2;
dnn11 = [0 * ones (1,sz2dne11) num11 * kv11 * kf11 0];
den11 = dne11 + dnn11;
// denominator of open loop joint transfer function with feedback controller
dimden11 = size (den11);
szden11 = dimden11 (2) - 1;
// order of denominator of joint 1
[s11,ns11] = sform (num11,den11);
gcnum11 = kp11 ;
// numerator of feedforward controller
```

```
gcden11 = 1;
// denominator of feedforward controller
gcgpnum11 = convolve (gcnum11,num11);
gcgpden11 = convolve (gcden11,den11);
[s,ns] = sform (gcgpnum11,gcgpden11);
[sl,nsl] = feedback (s,ns);
[tnum11,tden11] = tform (sl,nsl);
// numerator and denominator of closed loop transfer function with controller
tnum11 = gcgpnum11;
//
//
////////////////////////////////////////////////////////////////
/////
///// SYSTEM_BUILD CODE FOR SUPER-BLOCK BLOCK1
/////
////////////////////////////////////////////////////////////////
//
// The blocks of super-block BLOCK1 are defined in the following
// section
//
build;
Edit; block1; 0;
Define; 11;
Algebraic Equations; 2;
Block name; ser11; 3; 2; 4; 1;
Parameter Entry; 2; +1,-1;
Define; 12;
Gain block;
Block name; kp11; 3; 1; 4; 1;
Parameter entry; kp11;
Define; 2;
Dynamic Systems; 2;
Block name; ki11; 3; 1; 4; 1;
Parameter entry; 1; ki11; y;
Define; 13;
Algebraic Equations; 2;
Block name; sc11; 3; 1; 4; 1;
Parameter entry; 2; +1,+1;
Define; 14;
Dynamic Systems; 4;
Block name; gpkv11; 3; 1; 4; 1; 5; ns11;
Parameter entry; sznum11; num11; szden11; den11;
// The blocks of super-block BLOCK1 are connected in the following
```

```
// section
Connect;
External Input; 1; 11; 1,1; 0,0;
Internal Path; 11; 12;
Internal Path; 11; 2;
Internal Path; 12; 13; 1,2; 0,0;
Internal Path; 2; 13; 1,1; 0,0;
Internal Path; 13; 14;
Internal Path; 14; 11; 1,2; 0,0;
External Output; 1; 14;
top;
mat;
//
//
////////////////////////////////////////////////////////////////////////
/////
///// PARAMETERS OF SUPER-BLOCK T22 (FINGER 1 JOINT 2)
///// & SUPER-BLOCK BLOCK2
/////
////////////////////////////////////////////////////////////////////////
//
// Joint 2
//
kp22 = 8;
// proportional gain of PID controller
kv22 = 2;
// velocity gain of PID controller
ki22 = 0;
// integral gain of PID controller
kf22 = 1/122.43;
// gain of the differentiator block
n22 = 9955.55;
nm22 = 1;
num22 = convolve (n22,nm22);
// numerator of transfer function of joint 1 with feedback controller
d22 = [1 90 5625];
dn22 = 1;
dne22 = convolve (d22,dn22);
szdne22 = size (dne22);
sz2dne22 = szdne22 (2) - 2;
dd22 = [0 * ones (1,sz2dne22) num22 * kv22 * kf22 0];

den22 = dne22 + dd22;
// denominator of transfer function of joint 1 with feedback controller
```

```
gcnum22 = kp22;
// numerator of feedforward controller
gcden22 = 1;
// denominator of feedforward controller
gcgpnum22 = convolve (gcnum22,num22);
gcgpden22 = convolve (gcden22,den22);
[s22,ns22] = sform (num22,den22);
[st22,nst22] = sform (gcgpnum22,gcgpden22);
[stl22,nstl22] = feedback (st22,nst22);
[tnum22,tden22] = tform (stl22,nstl22);
tnum22 = gcgpnum22;
// numerator & denominator of closed loop transfer function of joint 2
// with controller.
pcool2num = [1 48 900];
pcool2den = [1 96 3600];
// numerator and denominator of output-output coupling between joint 1
// and joint 2.
tnum12 = convolve (tnum11,pcool2num);
tden12 = convolve (tden11,pcool2den);
// numerator and denominator of input-output coupling between joint 1
// and joint 2.
[s12,ns12] = sform (tnum12,tden12);
t22it12num = convolve (tden22,tnum12);
t22it12den = convolve (tnum22,tden12);
dimnum22 = size (num22);
sznum22 = dimnum22 (2) - 1;
dimden22 = size (den22);
szden22 = dimden22 (2) - 1;
[sp22,nsp22] = sform (num22,den22);
dimnum12 = size (tnum12);
sznum12 = dimnum12 (2) - 1;
dimden12 = size (tden12);
szden12 = dimden12 (2) - 1;
[s12,ns12] = sform (tnum12,tden12);
dimn22i12 = size (t22it12num);
szn22i12 = dimn22i12 (2) - 1;
dimd22i12 = size (t22it12den);
szd22i12 = dimd22i12 (2) - 1;
[s22i12,ns22i12] = sform (t22it12num,t22it12den);
```

```
////////////////////////////////////////////////////////////////
/////
///// SYSTEM_BUILD CODE FOR SUPER-BLOCK T22
/////
////////////////////////////////////////////////////////////////
//
// The blocks of super-block T22 are defined in the following
// section
build;
Edit; t22; 0;
Define; 11;
Algebraic Equations; 2;
Block name; ser22; 3; 2; 4; 1;
Parameter Entry; 2; +1,-1;
Define; 12;
Gain block;
Block name; kp22; 3; 1; 4; 1;
Parameter entry; kp22;
Define; 2;
Dynamic Systems; 2;
Block name; ki22; 3; 1; 4; 1;
Parameter entry; 1; ki22; y;
Define; 13;
Algebraic Equations; 2;
Block name; sc22; 3; 1; 4; 1;
Parameter entry; 2; +1,+1;
Define; 14;
Dynamic Systems; 4;
Block name; gpkv22; 3; 1; 4; 1; 5; nsp22;
Parameter entry; sznum22; num22; szden22; den22;
//
// The blocks of super-block T22 are defined in the following
// section
Connect;
External Input; 1; 11; 1,1; 0,0;
Internal Path; 11; 12;
Internal Path; 11; 2;
Internal Path; 12; 13; 1,2; 0,0;
Internal Path; 2; 13; 1,1; 0,0;
Internal Path; 13; 14;
Internal Path; 14; 11; 1,2; 0,0;
External Output; 1; 14;
top;
```

```
//
//
////////////////////////////////////////////////////////////////
/////
///// SYSTEM_BUILD CODE FOR SUPER-BLOCK BLOCK2
/////
////////////////////////////////////////////////////////////////
//
// The blocks of super-block BLOCK2 are defined in the
// following section
Edit Super-Block;
block2; 0
Define Block; 1;
Dynamic Systems; 4;
Block name; t22it12; 3; 1; 4; 1; 5; ns22i12;
Parameter Entry; szn22i12; t22it12num; szd22i12; t22it12den;
Define Block; 4;
Dynamic Systems; 4;
Block name; t12; 3; 1; 4; 1; 5; ns12;
Parameter Entry; sznum12; tnum12; szden12; tden12;
Define Block; 13;
Super-Block; Block name; t22; 3; 1; 4; 1;
Parameter Entry;
Define Block; 14;
Algebraic Equations; 2;
Block name; so12; 3; 2; 4; 1;
Parameter Entry; 2; +1,+1;
Define Block; 11;
Algebraic Equations; 2;
Block name; scr12; 3; 2; 4; 1;
Parameter entry; 2; +1,-1;
// The blocks of super-block BLOCK2 are connected in the
// following section
Connect;
External Input; 2; 1; 1,1; 0,0;
External Input; 2; 11; n; 2,1; 0,0;
External Input; 2; 4; 1,1; 0,0;
Internal Path; 1; 11; 1,2; 0,0;
Internal Path; 4; 14; 1,2; 0,0;
Internal Path; 13; 14; 1,1; 0,0;
Internal Path; 11; 13;
External Output; 1; 14;
top;
```

```
mat;
//
//
//
////////////////////////////////////////////////////////////////
/////
///// PARAMETERS OF SUPER-BLOCK T33 (FINGER 1 JOINT 3),
///// SUPER-BLOCK BLOCK3 AND SUPER-BLOCK BLOCK4
/////
////////////////////////////////////////////////////////////////
//
// The values of the different blocks of super-blocks T33,
// BLOCK3 and BLOCK4 are calculated in this section
kp33 = 8;
// proportional gain of PID controller
kv33 = 2;
// velocity gain of PID controller
ki33 = 0;
// integral gain of PID controller
kf33 = 1/120.46;
// gain of differentiator
nm33 = 17787.94;
// numerator of open loop transfer function of joint 3
// with feedback controller
n33 = 1;
num33 = convolve (n33,nm33);
dimnum33 = size (num33);
sznum33 = dimnum33 (2) - 1;
dn33 = [1 340 22500];
d33 = 1;
dne33 = convolve (dn33,d33);
szdne33 = size (dne33);
sz2dne33 = szdne33 (2) - 2;
dd33 = [0 * ones (1,sz2dne33) num33 * kv33 * kf33 0];

den33 = dne33 + dd33;
// denominator of open loop transfer function of joint 3
// with feedback controller
dimden33 = size (den33);
szden33 = dimden33 (2) - 1;
[sp33,nsp33] = sform (num33,den33);
gc33num =kp33; // numerator of feedforward controller
gc33den = 1; // denominator of feedforward controller
gcgp33num = convolve (gc33num,num33);
```

```
gcgp33den = convolve (gc33den,den33);
[scp33,nscp33] = sform (gcgp33num,gcgp33den);
[st33,nst33] = feedback (scp33,nscp33);
[tnum33,tden33] = tform (st33,nst33);
// numerator and denominator of closed loop transfer function of joint 3
// with controller
tnum33 = gcgp33num;
pcoo13num = 643.383;
pcoo13den = [1 18 8100];
// numerator and denominator of coupling between
// outputs of joint 1 & joint 3
tnum13 = convolve (tnum11,pcoo13num);
tden13 = convolve (tden11,pcoo13den);
// numerator and denominator of coupling between
// outputs of joint 1 & joint 3
pcoo23num = .9576*[1 70 2500];
pcoo23den = [1 119 7225];
tnum23 = convolve (tnum22,pcoo23num);
tden23 = convolve (tden22,pcoo23den);
t33it13num = convolve (tden33,tnum13);
t33it13den = convolve (tnum33,tden13);
[s33i13,ns33i13] = sform (t33it13num,t33it13den);
t33it23num = convolve (tden33,tnum23);
t33it23den = convolve (tnum33,tden23);
[s33i23,ns33i23] = sform (t33it23num,t33it23den);
[st13,nst13] = sform (tnum13,tden13);
[st23,nst23] = sform (tnum23,tden23);
dimnum13 = size (tnum13);
sznum13 = dimnum13 (2) - 1;
dimden13 = size (tden13);
szden13 = dimden13 (2) - 1;
dimnum23 = size (tnum23);
sznum23 = dimnum23 (2) - 1;
dimden23 = size (tden23);
szden23 = dimden23 (2) - 1;
dimn33i13 = size (t33it13num);
szn33i13 = dimn33i13 (2) - 1;
dimd33i13 = size (t33it13den);
szd33i13 = dimd33i13 (2) - 1;
dimn33i23 = size (t33it23num);
szn33i23 = dimn33i23 (2) - 1;
dimd33i23 = size (t33it23den);
szd33i23 = dimd33i23 (2) - 1;
```

```
////////////////////////////////////////////////////////////
/////
///// SYSTEM_BUILD CODE FOR SUPER-BLOCK T33
/////
////////////////////////////////////////////////////////////
//
// The blocks of super-block T33 are defined
// in the following section
build;
Edit; t33; 0;
Define; 11;
Algebraic Equations; 2;
Block name; ser33; 3; 2; 4; 1;
Parameter Entry; 2; +1,-1;
Define; 12;
Gain block;
Block name; kp33; 3; 1; 4; 1;
Parameter entry; kp33;
Define; 2;
Dynamic Systems; 2;
Block name; ki33; 3; 1; 4; 1;
Parameter entry; 1; ki33; y;
Define; 13;
Algebraic Equations; 2;
Block name; sc33; 3; 1; 4; 1;
Parameter entry; 2; +1,+1;
Define; 14;
Dynamic Systems; 4;
Block name; gpkv33; 3; 1; 4; 1; 5; nsp33;
Parameter entry; sznum33; num33; szden33; den33;
Connect;
External Input; 1; 11; 1,1; 0,0;
Internal Path; 11; 12;
Internal Path; 11; 2;
Internal Path; 12; 13; 1,2; 0,0;
Internal Path; 2; 13; 1,1; 0,0;
Internal Path; 13; 14;
Internal Path; 14; 11; 1,2; 0,0;
External Output; 1; 14;
top;
```

```
////////////////////////////////////////////////////////////////
/////
///// SYSTEM_BUILD CODE FOR SUPER-BLOCK BLOCK4
/////
////////////////////////////////////////////////////////////////
//
// The blocks of super-block BLOCK4 are defined in the
// following section
Edit Super-Block;
block4; 0;
Define Block; 1;
Dynamic Systems; 4;
Block name; t13; 3; 1; 4; 1; 5; nst13;
Parameter Entry; sznum13; tnum13; szden13; tden13;
Define Block; 11;
Dynamic Systems; 4;
Block name; t23; 3; 1; 4; 1; 5; nst23;
Parameter Entry; sznum23; tnum23; szden23; tden23;
Define Block; 21;
Super-Block;
Block name; t33; 3; 1; 4; 1;
Parameter Entry;
Define Block; 23;
Algebraic Equations; 2;
Block name; so23; 3; 3; 4; 1;
Parameter Entry; 3; +1,+1,+1;
// The blocks of super-block BLOCK4 are
// connected in the following section
Connect Blocks;
External Input; 3; 1; 1,1; 0,0;
External Input; 3; 11; 2,1; 0,0;
External Input; 3; 21; 3,1; 0,0;
Internal Path; 1; 23; 1,1; 0,0;
Internal Path; 11; 23; 1,2; 0,0;
Internal Path; 21; 23; 1,3; 0,0;
External Output; 1; 23;
top;
```

```
////////////////////////////////////////////////////////////////
/////
///// SYSTEM_BUILD CODE FOR SUPER-BLOCK BLOCK3
/////
////////////////////////////////////////////////////////////////
//
// The blocks of BLOCK3 are defined in the following section
Edit Super-Block;
block3; 0;
Define Block; 4;
Dynamic Systems; 4;
Block name; t33t13; 3; 1; 4; 1; 5; ns33i13;
Parameter Entry; szn33i13; t33it13num; szd33i13; t33it13den;
Define Block; 1;
Dynamic Systems; 4;
Block name; t33t23; 3; 1; 4; 1; 5; ns33i23;
Parameter Entry; szn33i23; t33it23num; szd33i23; t33it23den;
Define Block; 13;
Algebraic Equations; 2;
Block name; ser123; 3; 3; 4; 1;
Parameter Entry; 3; +1,-1,-1;
// The blocks of super-block BLOCK3 are defined
// in the following section
Connect;
External Input; 3; 13; n; 3,1; 0,0;
External Input; 3; 1; 2,1; 0,0;
External Input; 3; 4; 1,1; 0,0;
Internal Path; 1; 13; 1,2; 0,0;
Internal Path; 4; 13; 1,3; 0,0;
External Output; 1; 13;
top;
//
//
////////////////////////////////////////////////////////////////
/////
///// SYSTEM_BUILD CODE FOR SUPER-BLOCK DEC
/////
////////////////////////////////////////////////////////////////
//
// The blocks of super-block DEC are defined in the
// following section
Edit Super-Block;
dec; 0;
```

```
Define Block; 2;
Super-Block;
Block name; block1; 3; 1; 4; 1;
Parameter Entry;
Define Block; 12;
Super-Block;
Block name; block2; 3; 2; 4; 1;
Parameter Entry;
Define Block; 21;
Super-Block;
Block name; block3; 3; 3; 4; 1;
Parameter Entry;
Define Block; 22;
Super-Block;
Block name; block4; 3; 3; 4; 1;
Parameter Entry;
Connect Blocks;
External Input; 3; 2; 1,1; 0,0;
External Input; 3; 12; 1,1; 2,2; 0,0;
External Input; 3; 21; n; 1,1; 2,2; 3,3; 0,0;
External Input; 3; 22; n; 1,1; 2,2; 0,0;
Internal Path; 21; 22; 1,3; 0,0;
External Output; 3; 2; 1,1; 0,0;
External Output; 3; 12; 1,2; 0,0;
External Output; 3; 22; 1,3; 0,0;
top;
mat;
save 'spblocks';
// saves all the variables and the super-blocks in the file 'spblocks'
```

```
////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////////
//////
//////                    PROGRAM MODPROGBLS
////// The function of this program is to load the simulation segments into
////// MATRIX_x for modifying the same by changing the various
////// parameters of the models of the different joints.
//////
////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////////
//
//
// Loading of simulation segments.
load 'spblocks'
// Loads super-blocks BLOCK1, BLOCK2, BLOCK3, BLOCK4, T22, T33, DEC
//
//
////////////////////////////////////////////////////////////////////
/////
///// Parameters of super-block BLOCK1
/////
////////////////////////////////////////////////////////////////////
//
//
kp11 = 8;
// proportional gain of PID controller of finger 1 joint 1
kv11 = 2;
// derivative gain of PID controler of finger 1 joint 1
ki11 = 1;
// integral gain of finger 1 joint 1
kf11 = 1/124.31;
// gain of the differentiator of finger 1 joint 1
n11 = 3450 * 12.5/4;
nm11 = 1;
num11 = convolve (n11,nm11); // numerator of plant with velocity feedback
dimnum11 = size (num11);
sznum11 = dimnum11 (2) - 1;
// order of numerator of joint 1
d11 = [1 245 3450];
dn11 = 1;
dne11 = convolve (d11,dn11);
szdne11 = size (dne11);
sz2dne11 = szdne11 (2) - 2;
```

```
dnn11 = [0 * ones (1,sz2dne11) num11 * kv11 * kf11 0];
den11 = dne11 + dnn11;
// denominator of open loop joint transfer function with feedback controller
[s11,ns11] = sform (num11,den11);
gcnum11 = [kp11 ki11];
// numerator of feedforward controller
gcden11 = [1 0];
// denominator of feedforward controller
gcgpnum11 = convolve (gcnum11,num11);
gcgpden11 = convolve (gcden11,den11);
[s,ns] = sform (gcgpnum11,gcgpden11);
[sl,nsl] = feedback (s,ns);
[tnum11,tden11] = tform (sl,nsl);
// numerator and denominator of closed loop transfer function with controller
tnum11 = gcgpnum11;
//
//
////////////////////////////////////////////////////////////////////
/////
///// Parameters of super-block T22
/////
////////////////////////////////////////////////////////////////////
//
//
kp22 = 8;
// proportional gain of PID controller
kv22 = 2;
// velocity gain of PID controller
ki22 = 1;
// integral gain of PID controller
kf22 = 1/122.43;
// gain of the differentiator
n22 = 9955.55;
nm22 = 1;
num22 = convolve (n22,nm22);
// numerator of transfer function of joint 1 with feedback controller
d22 = [1 90 5625];
dn22 = 1;
dne22 = convolve (d22,dn22);
szdne22 = size (dne22);
sz2dne22 = szdne22 (2) - 2;
dd22 = [0 * ones (1,sz2dne22) num22 * kv22 * kf22 0];
den22 = dne22 + dd22;
```

```
// denominator of transfer function of joint 1 with feedback controller
//
//
////////////////////////////////////////////////////////////////////
/////
///// Parameters of super-block BLOCK2
/////
////////////////////////////////////////////////////////////////////
//
//
[s22,ns22] = sform (num22,den22);
[st22,nst22] = sform (gcgpnum22,gcgpden22);
[stl22,nstl22] = feedback (st22,nst22);
[tnum22,tden22] = tform (stl22,nstl22);
tnum22 = gcgpnum22;
// numerator & denominator of closed loop transfer function of joint 2
// with controller.
pcoo12num = [1 48 900];
// numerator of coupling between outputs of joints 1 & 2
pcoo12den = [1 96 3600];
// denominator of coupling between outputs of joints 1 & 2
tnum12 = convolve (tnum11,pcoo12num);
tden12 = convolve (tden11,pcoo12den);
// numerator and denominator of input-output coupling between joint 1
// and joint 2.
[s12,ns12] = sform (tnum12,tden12);
t22it12num = convolve (tden22,tnum12);
t22it12den = convolve (tnum22,tden12);
dimnum22 = size (num22);
sznum22 = dimnum22 (2) - 1; // order of numerator of joint 2
[sp22,nsp22] = sform (num22,den22);
dimnum12 = size (tnum12);
sznum12 = dimnum12 (2) - 1;
[s12,ns12] = sform (tnum12,tden12);
dimn22i12 = size (t22it12num);
szn22i12 = dimn22i12 (2) - 1;
[s22i12,ns22i12] = sform (t22it12num,t22it12den);
[s11,ns11] = sform (num11,den11);
dimnum11 = size (num11);
sznum11 = dimnum11 (2) - 1;
gcnum11 = [kp11 ki11];
gcden11 = [1 0];
gcgpnum11 = convolve (gcnum11,num11);
```

```
gcgpden11 = convolve (gcden11,den11);
[s,ns] = sform (gcgpnum11,gcgpden11);
//
//
//
////////////////////////////////////////////////////////////////////////
//////
///// Parameters of super-block T33, super-block BLOCK3 and
///// super-block BLOCK4 ( finger 1 joint 3)
/////
////////////////////////////////////////////////////////////////////////
//
// The values of the different blocks of super-blocks T33, BLOCK3 and
// BLOCK4 are calculated in this section
kp33 = 8;
// proportional gain of PID controller
kv33 = 2;
// velocity gain of PID controller
ki33 = 1;
// integral gain of PID controller
kf33 = 1/120.46;
// gain of differentiator
nm33 = 17787.94;
n33 = 800;
num33 = convolve (n33,nm33);
// numerator of open loop transfer function of joint 3
// with feedback controller
dimnum33 = size (num33);
sznum33 = dimnum33 (2) - 1;
dn33 = [1 340 22500];
d33 = [1 800];
dne33 = convolve (dn33,d33);
szdne33 = size (dne33);
sz2dne33 = szdne33 (2) - 2;
dd33 = [0 * ones (1,sz2dne33) num33 * kv33 * kf33 0];
den33 = dne33 + dd33;
//denominator of open loop transfer function of joint 3 with feedback controller
[sp33,nsp33] = sform (num33,den33);
gc33num =[kp33 ki33];
// numerator of feedforward controller
gc33den = [1 0];
// denominator of feedforward controller
gcgp33num = convolve (gc33num,num33);
```

```
gcgp33den = convolve (gc33den,den33);
[scp33,nscp33] = sform (gcgp33num,gcgp33den);
[st33,nst33] = feedback (scp33,nscp33);
[tnum33,tden33] = tform (st33,nst33);
// numerator and denominator of closed loop transfer function of joint 3
// with controller
tnum33 = gcgp33num;
pcoo13num = 643.383;
// numerator and denominator of coupling between
pcoo13den = [1 18 8100];
// outputs of joint 1 & joint 3
tnum13 = convolve (tnum11,pcoo13num);
// numerator & denominator of coupling
tden13 = convolve (tden11,pcoo13den);
// between input of joint 1 and
// output of joint 3
pcoo23num = .9576 * [1 70 2500];
pcoo23den = [1 119 7225];
// numerator and denominator of coupling between outputs of joint 1
// and joint 3
tnum23 = convolve (tnum22,pcoo23num);
tden23 = convolve (tden22,pcoo23den);
// numerator and denominator of coupling between input of joint 2
// and output of joint 3
t33it13num = convolve (tden33,tnum13);
t33it13den = convolve (tnum33,tden13);
[s33i13,ns33i13] = sform (t33it13num,t33it13den);
t33it23num = convolve (tden33,tnum23);
t33it23den = convolve (tnum33,tden23);
[s33i23,ns33i23] = sform (t33it23num,t33it23den);
[st13,nst13] = sform (tnum13,tden13);
[st23,nst23] = sform (tnum23,tden23);
dimnum13 = size (tnum13);
sznum13 = dimnum13 (2) - 1;
dimnum23 = size (tnum23);
sznum23 = dimnum23 (2) - 1;
dimn33i13 = size (t33it13num);
szn33i13 = dimn33i13 (2) - 1;
dimn33i23 = size (t33it23num);
szn33i23 = dimn33i23 (2) - 1;
//
//
////////////////////////////////////////////////////////
```

```
/////
/////          Editing super-block BLOCK1
/////
////////////////////////////////////////////////////////
//
//
Edit; block1; 0;
Modify; 12;
New Block Parameters; kp11;
Done Update Block;
Modify; 2;
New Block Parameters; 1; ki11;
Done Update Block;
Modify; 14;
New Block Parameters; sznum11; num11; szden11; den11;
Done Update Block; top; mat;
//
//
////////////////////////////////////////////////////////
/////
///// Editing super-block T22
/////
////////////////////////////////////////////////////////
//
//
Build;
Edit; t22; 0;
Modify; 12;
New Block Parameters; kp22;
Done Update Block;
Modify; 2;
New Block Parameters; 1; ki22;
Done Update Block;
Modify; 14;
New Block Parameters; sznum22; num22; szden22; den22;
Done Update Block;
top
//
//
////////////////////////////////////////////////////////
/////
///// Editing super-block BLOCK2
/////
```

```
////////////////////////////////////////////////////
//
//
Edit; block2; 0
Modify; 1;
New Block Parameters; szn22i12; t22it12num; t22it12den;
Done Update Block;
Modify; 2;
New Block Parameters; sznum12; tnum12; szden12; tden12; 6;
//
//
////////////////////////////////////////////////////
/////
///// Editing super-block T33
/////
////////////////////////////////////////////////////
//
//
build;
Edit; t33; 0;
Modify; 12;
New Block Parameters; kp33;
Done Update Block;
Modify; 2;
New Block Parameters; 1; ki33;
Done Update Block;
Modify; 14;
New Block Parameters; sznum33; num33; den33;
Done Update Block;
top;
//
//
////////////////////////////////////////////////////
/////
///// Editing super-block BLOCK4
/////
////////////////////////////////////////////////////
//
//
Edit; block4; 0;
Modify; 1;
New Block Parameters; sznum13; tnum13; tden13;
Done Update Block;
```

```
Modify; 11;
New Block Parameters; sznum23; tnum23; tden23;
Done Update Block;
//
//
////////////////////////////////////////////////
/////
///// Editing super-block BLOCK3
/////
////////////////////////////////////////////////
//
//
Edit; block3; 0;
Modify; 4;
New Block Parameters; szn33i13; t33it13num; t33it13den;
Done Update Block;
Modify; 1;
New Block Parameters; szn33i23; t33it23num; t33it23den;
Done Update Block;
top;
//
//
save 'varspblocks';
// This command saves all the variables created in a file called
// varspblocks.
```

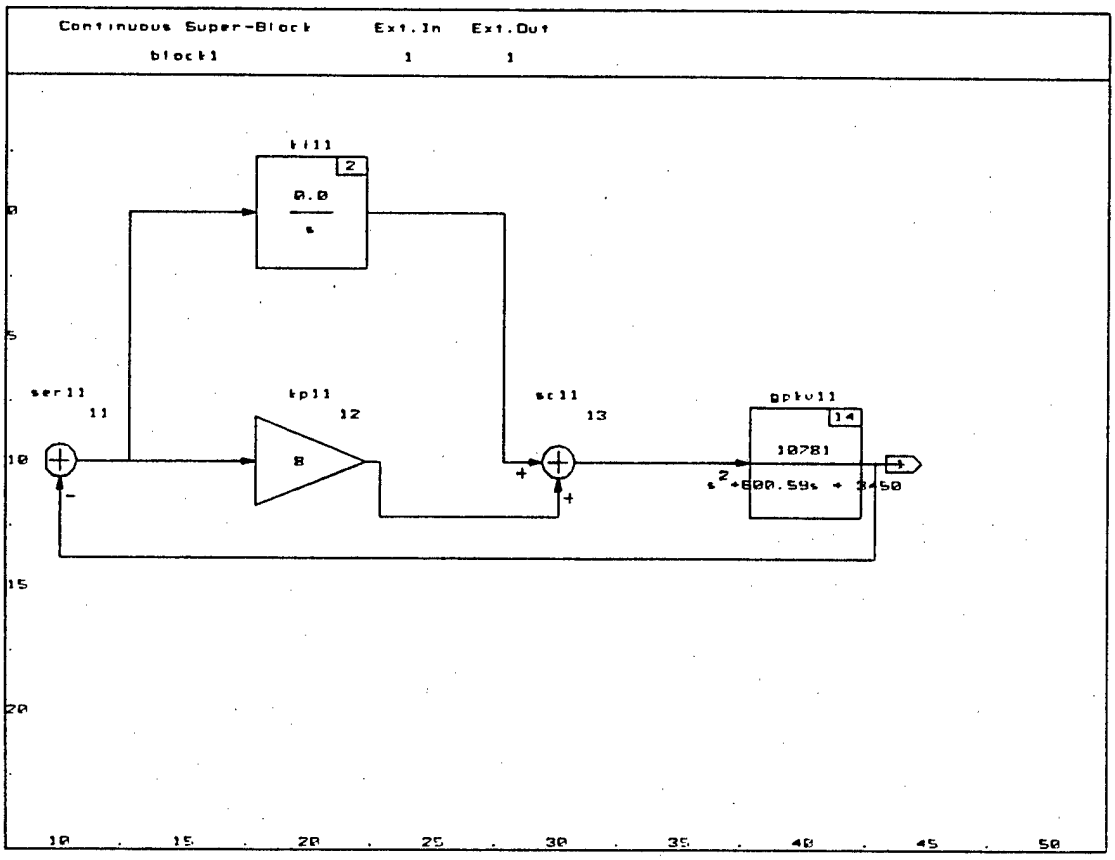



**Figure G1 Position control loop of finger 1 joint 1**

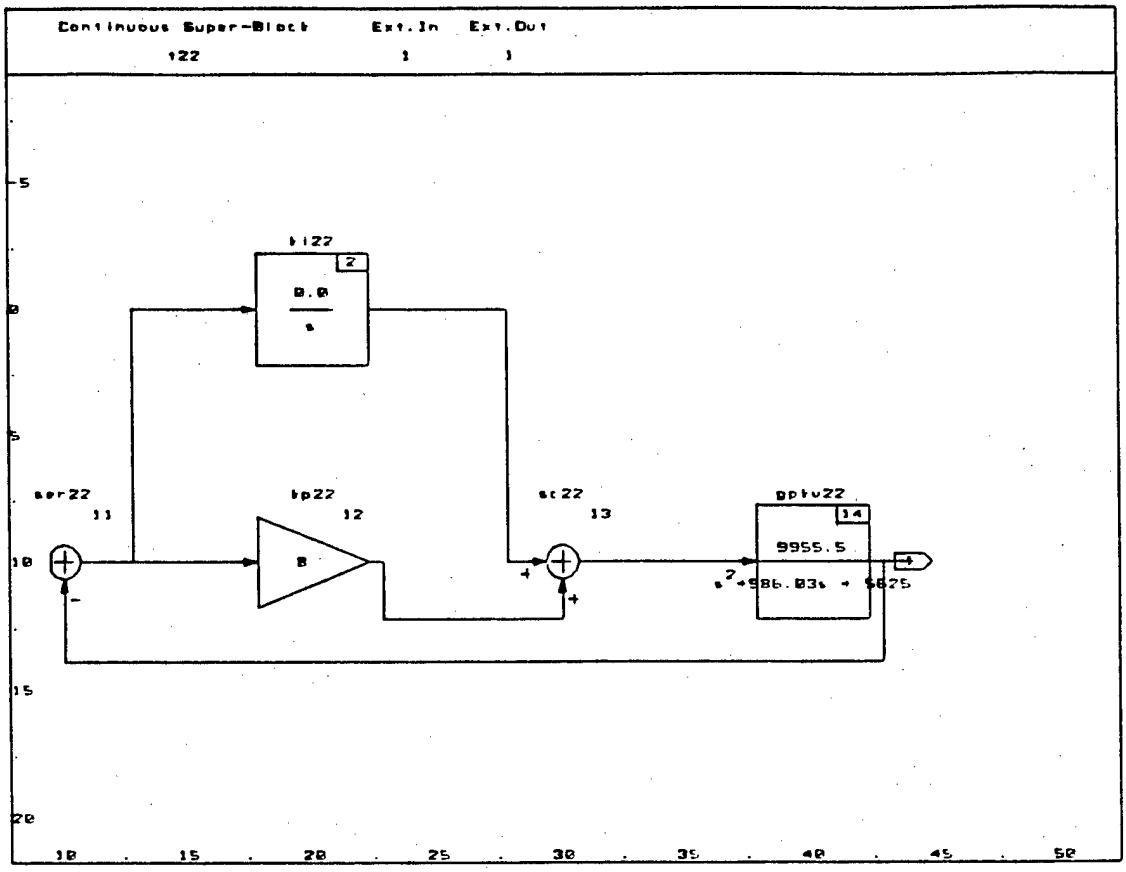



**Figure G2 Position control loop of finger 1 joint 2**

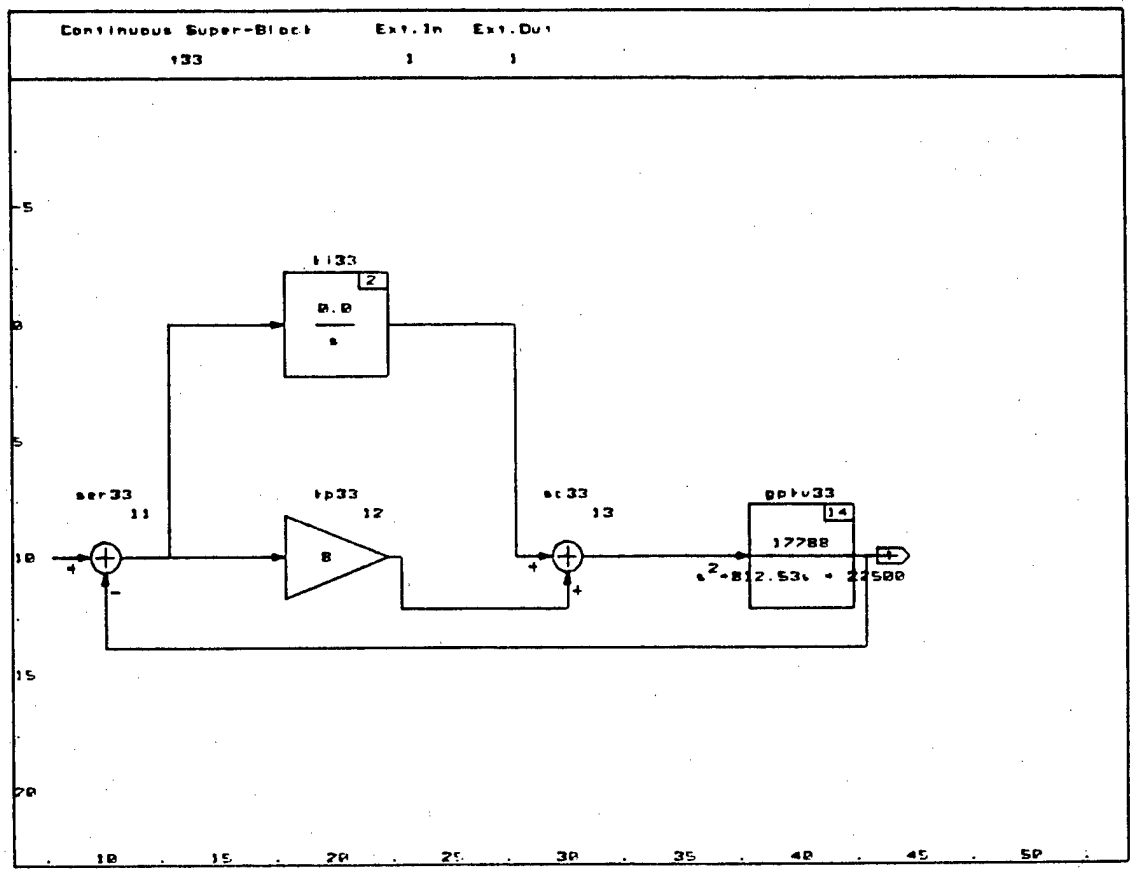



**Figure G3 Position control loop of finger 1 joint 3**

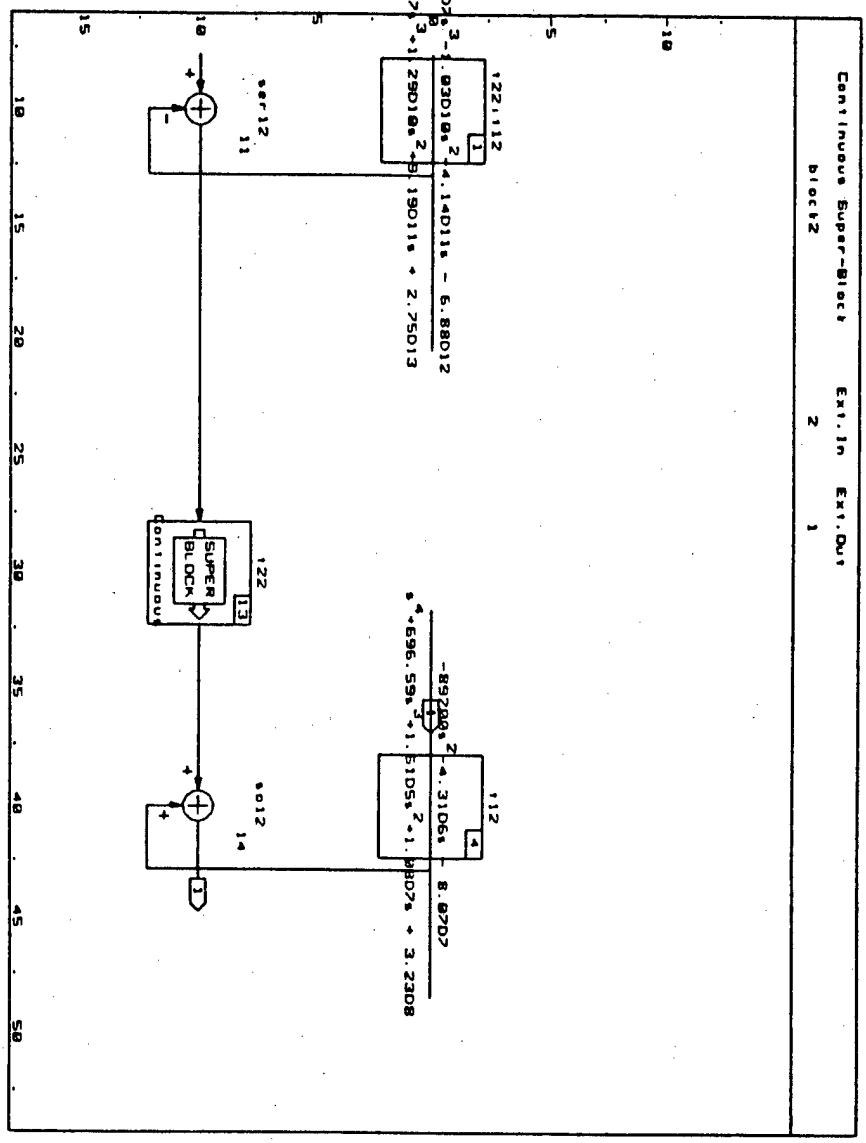



Figure G4 Coupling between finger 1 joints 1 and 2

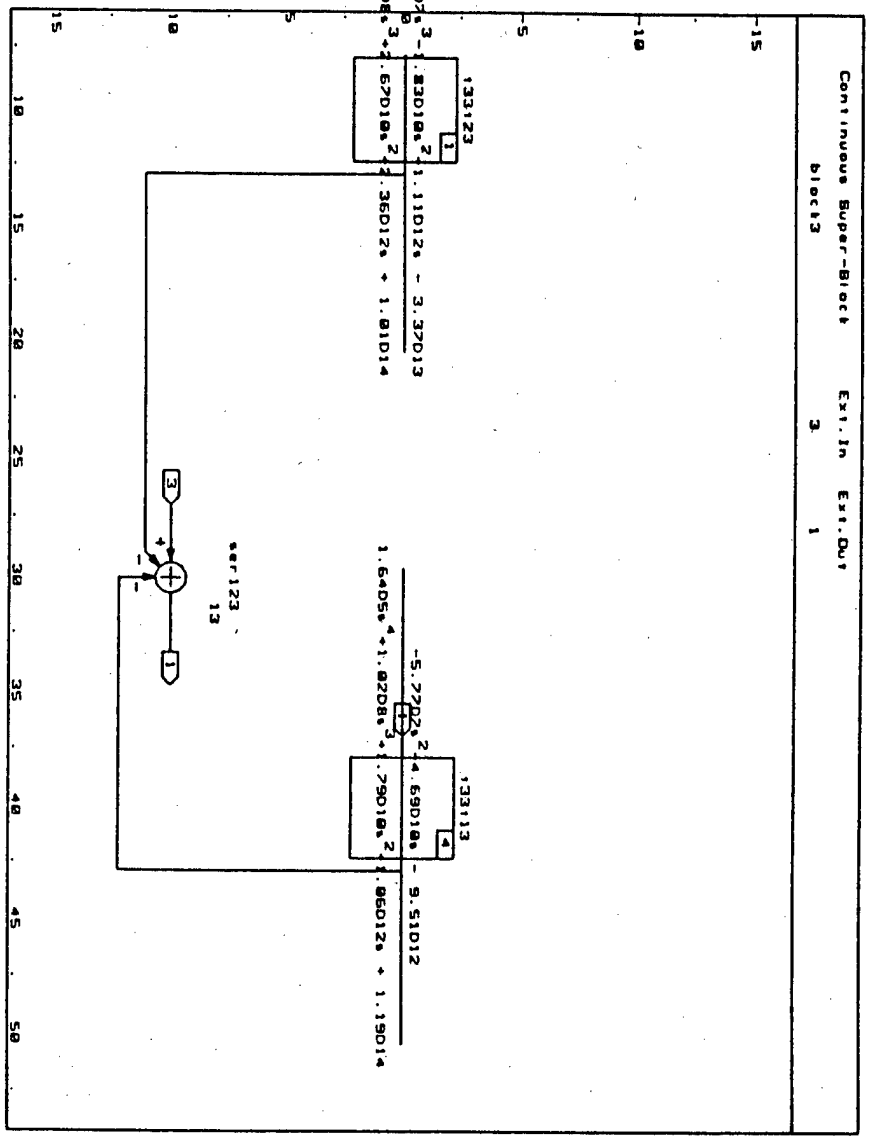



Figure G5 Coupling between finger 1 joint 3 and joints 1 and 2

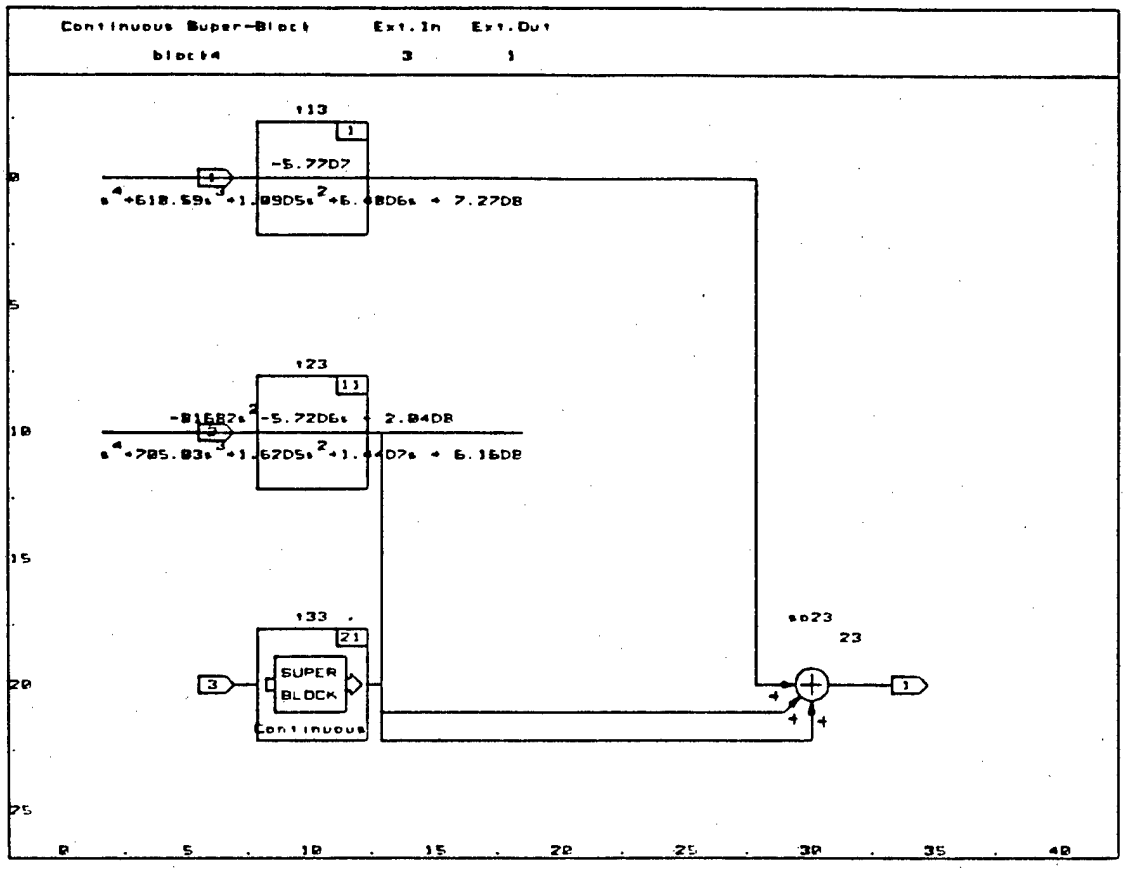



**Figure G6 Coupling between finger 1 joint 3 and joints 1 and 2**

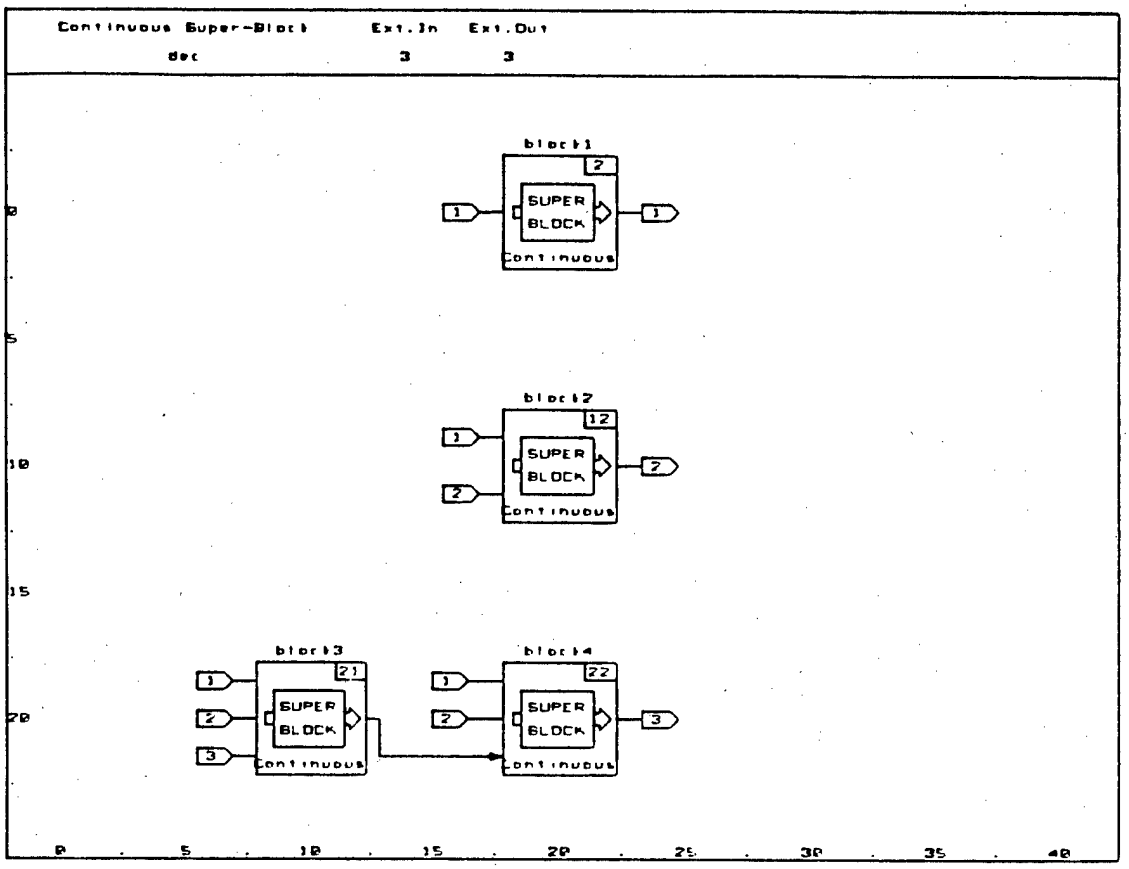



**Figure G7 Finger 1 joints 1, 2 and 3**

# REFERENCES


1. Working paper from AAMRL.

2. John M. Hollerbach, "Workshop on the Design and Control of Dextrous Hands", AI Lab Memo No. 661, Massachussetts Instittue of Technology, April 1982.

3. Crossley F.R.F. and Umholtz F. G., "Design for a three fingered Hand - Mechanism and Machine Theory", 1977.

4. S.C. Jacobsen, E.K. Iversen, D.F. Knutti, R.T. Johnson, K.B. Biggers, "Design of the UTAH/MIT Dextrous Hand", IEEE International Conference on Robotics and Automation, April 1986.

5. K. B. Biggers, S.C. Jacobsen, G. E. Gerpheide, "Low level control of the Utah/MIT dextrous hand", IEEE International Conference on Robotics and Automation, San Francisco, California, 1986.

6. Benjamin C. Kuo, "Automatic Control Systems", PRENTICE-HALL, INC., Englewood Cliffs, NJ 07632.

7. S.C. Jacobsen, J.E. Wood, D.F. Knutti, K.B. Biggers, "The Utah/M.I.T. Dexterous Hand : Work in Progress", The International Journal of Robotics Research, vol. 4, no.3, pp 21-50 (Winter 1984).

8. C.S.G. Lee, R.C. Gonzalez, K.S. Fu, " Tutorial on Robotics", IEEE Computer Society Press, 1983.

9. Brian W. Kernighan and Dennis M. Ritchie, " The C Programming Language", Second Edition, PRENTICE-HALL, INC., Englewood Cliffs, NJ 07632, 1988.

10. John J. Craig, " Introduction to Robotics - Mechanisms and Control", Addison-Wesley Publishing Co., Reading, Massachussetts, 1986.

11. "MATRIX$_x$ User's Guide", Integrated Systems, Inc. 1986.